Computational Optimal Transport

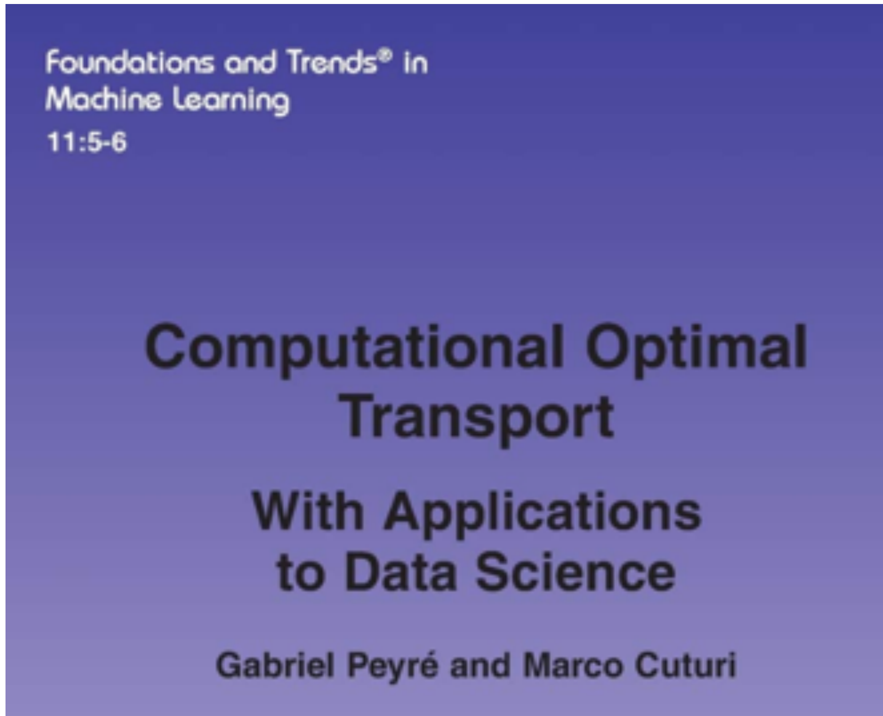
Marco Cuturi

Google AI Brain Team

ENSAE ParisTech
École nationale de la statistique et de l'administration économique
université PARIS-SACLAY

Found. & Trends in ML
survey with Gabriel Peyré

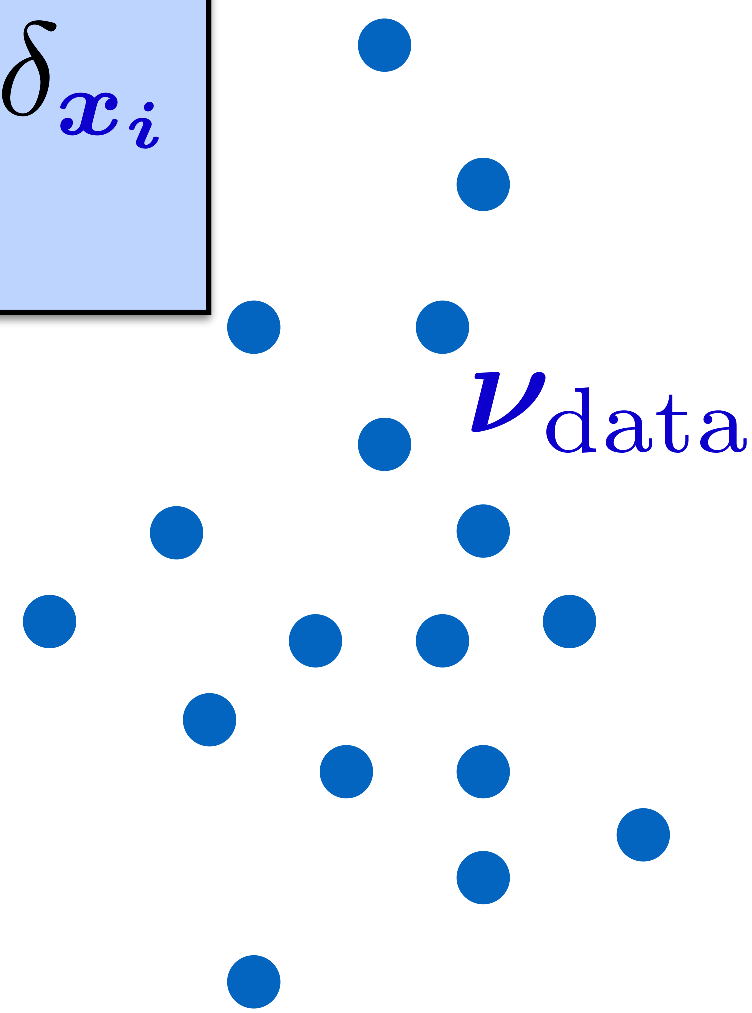
https://optimaltransport.github.io/



A Motivating Example

We collect data

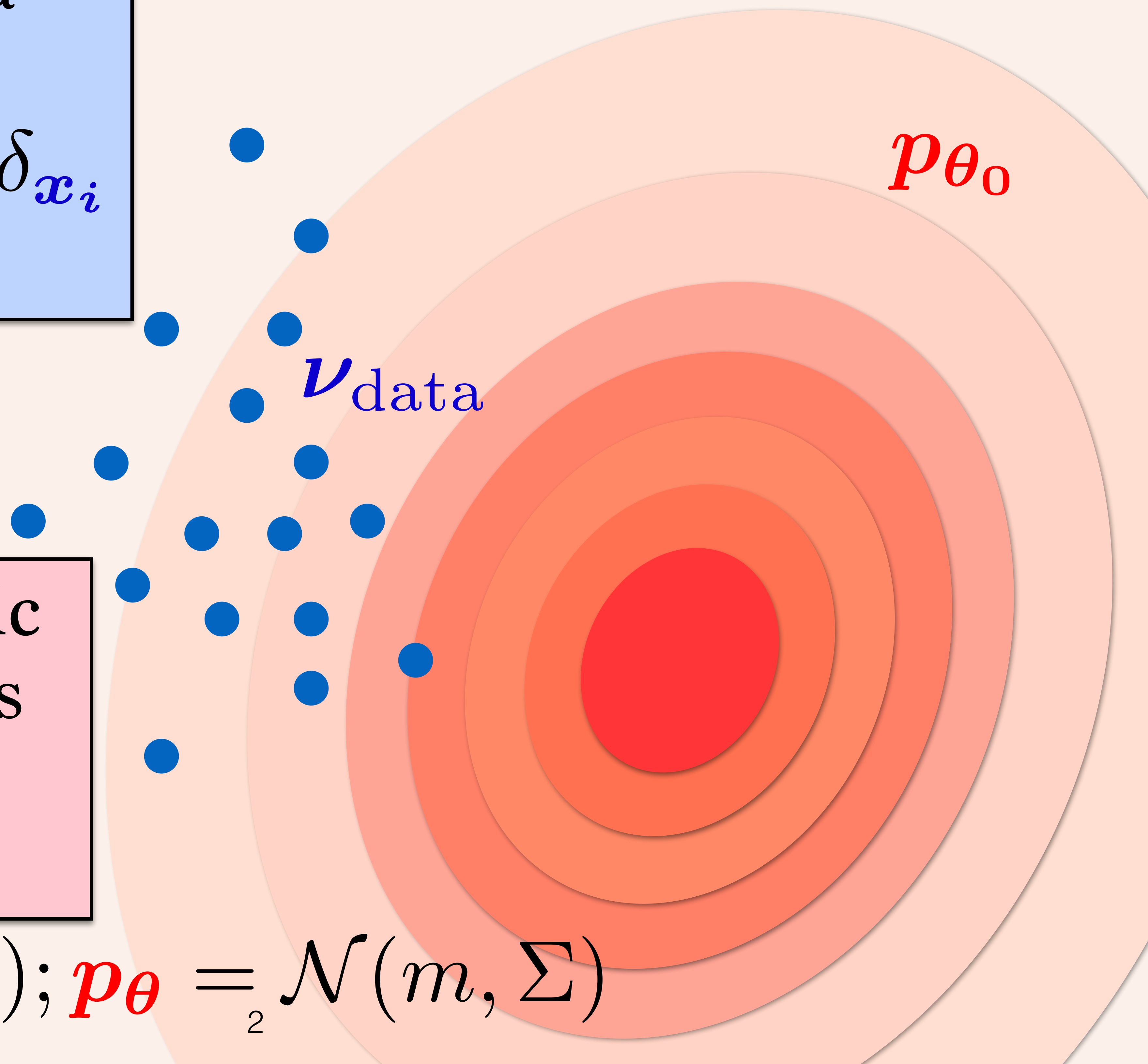
$$\boldsymbol{\nu}_{\text{data}} = \frac{1}{N}\sum_{i=1}^{N} \delta_{\boldsymbol{x_i}}$$



A Motivating Example

We collect data

$$\boldsymbol{\nu}_{\rm data} = \frac{1}{N}\sum_{i=1}^{N}\delta_{\boldsymbol{x}_{\boldsymbol{i}}}$$

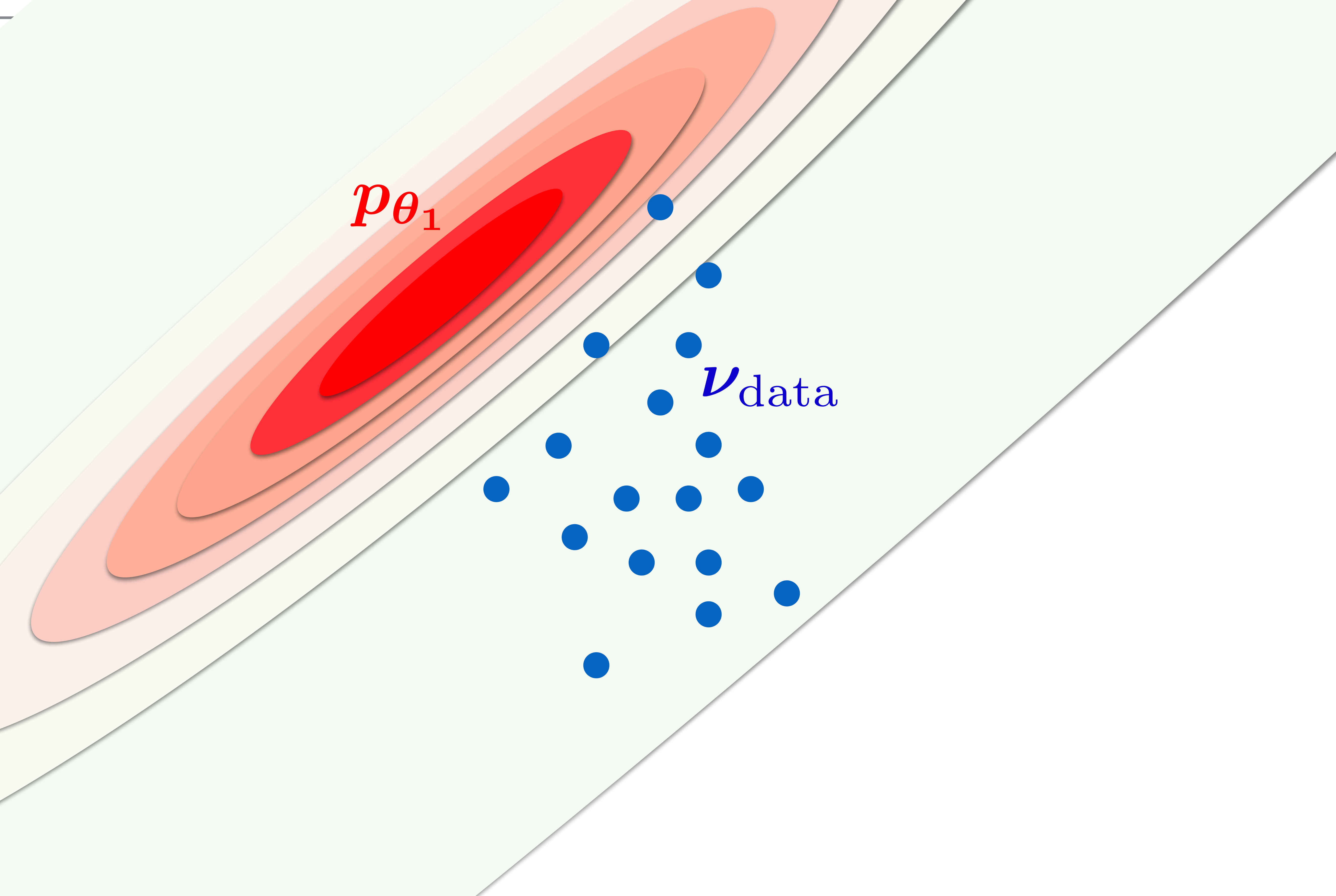


We fit a parametric family of densities

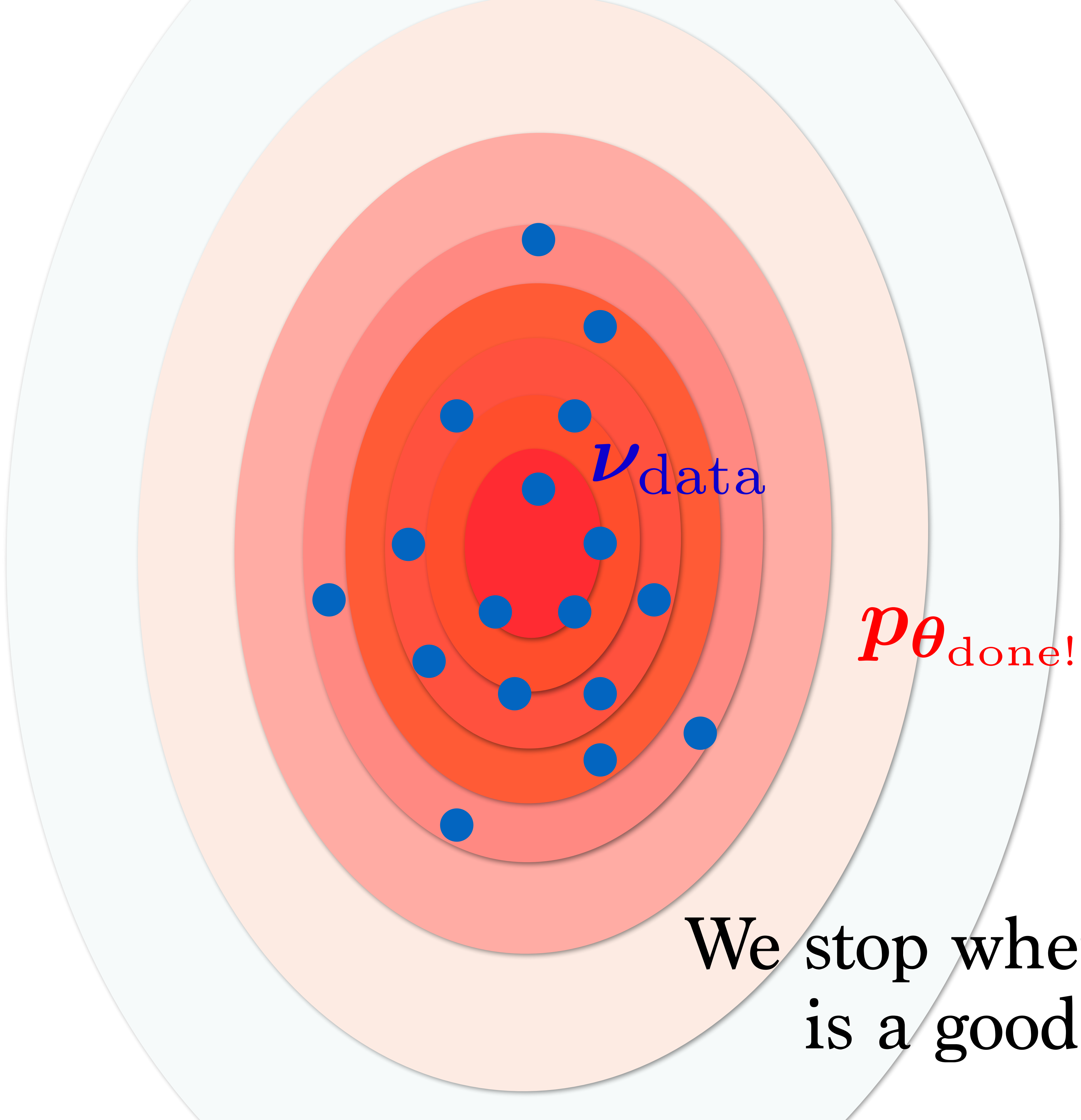
$$\{p_{\boldsymbol{\theta}}, \boldsymbol{\theta} \in \boldsymbol{\Theta}\}$$

$$e.g.\;\; \boldsymbol{\theta} = (m, \Sigma); \boldsymbol{p}_{\boldsymbol{\theta}} = \mathcal{N}(m, \Sigma)$$

Statistics 0.1: Density Fitting

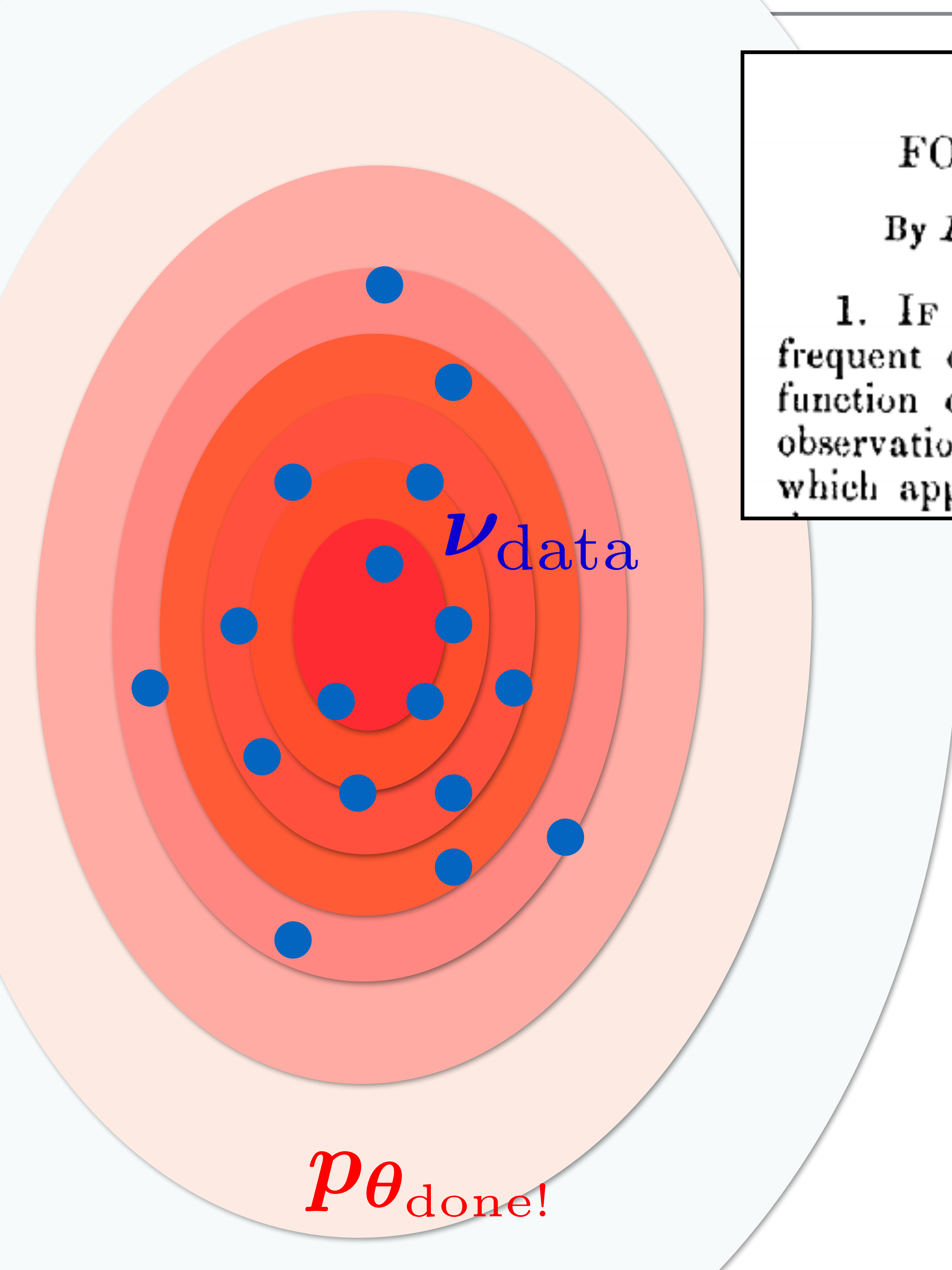


Statistics 0.1: Density Fitting



We stop when there is a good fit.

Maximum Likelihood Estimation



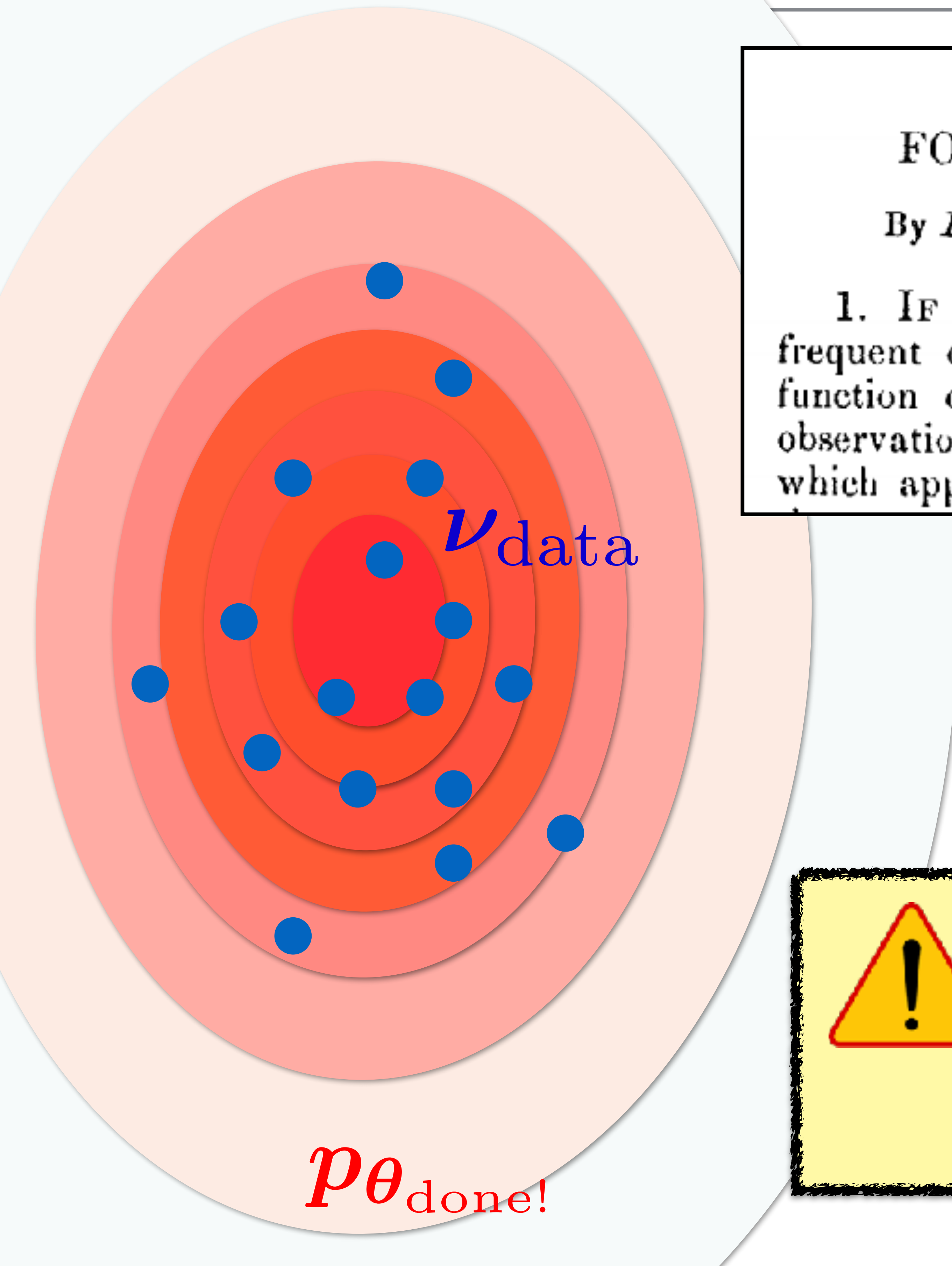
ON AN ABSOLUTE CRITERION
FOR FITTING FREQUENCY CURVES.

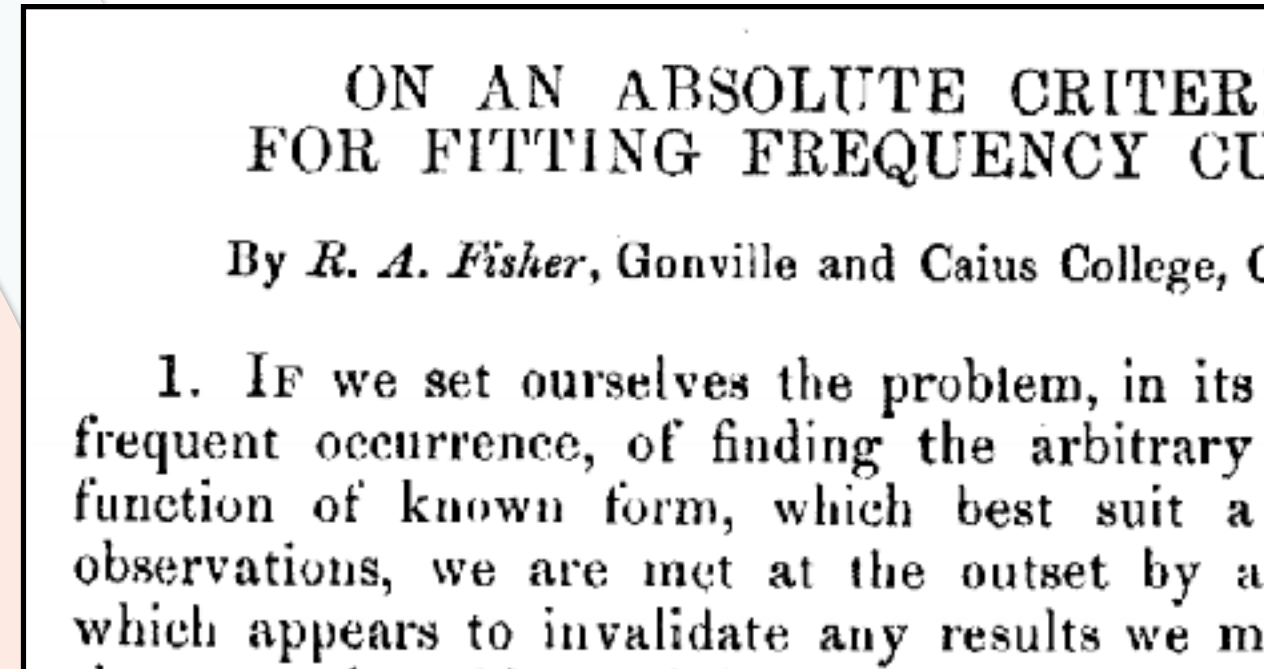
By *R. A. Fisher*, Gonville and Caius College, Cambridge.

1. If we set ourselves the problem, in its
frequent occurrence, of finding the arbitrary
function of known form, which best suit a
observations, we are met at the outset by an
which appears to invalidate any results we ma

$$\max_{\boldsymbol{\theta}\in\Theta} \frac{1}{N}\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}_i)$$

Maximum Likelihood Estimation





ON AN ABSOLUTE CRITERION
FOR FITTING FREQUENCY CURVES.

By *R. A. Fisher*, Gonville and Caius College, Cambridge.

1. If we set ourselves the problem, in its
frequent occurrence, of finding the arbitrary
function of known form, which best suit a
observations, we are met at the outset by an
which appears to invalidate any results we ma

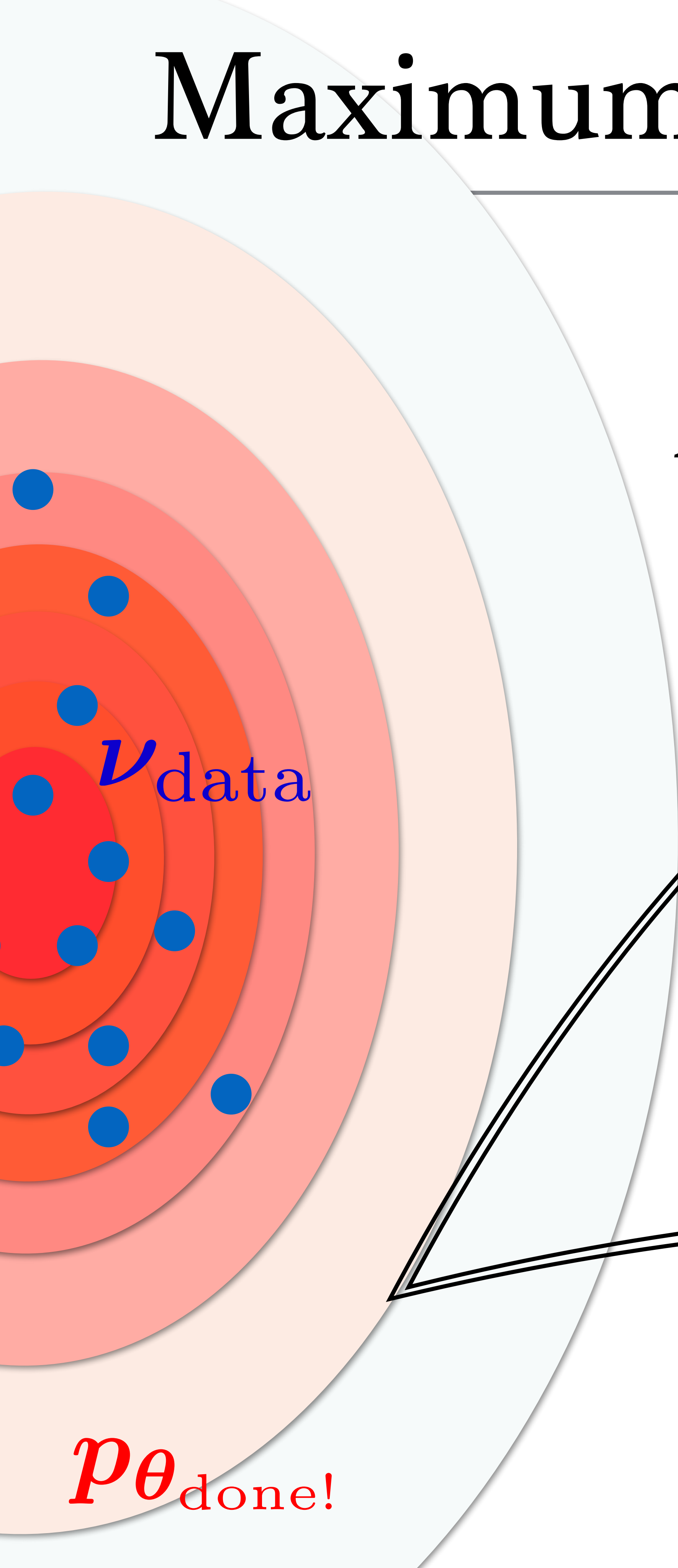
$$\max_{\theta \in \Theta} \frac{1}{N} \sum_{i=1}^{N} \log p_{\theta}(x_i)$$

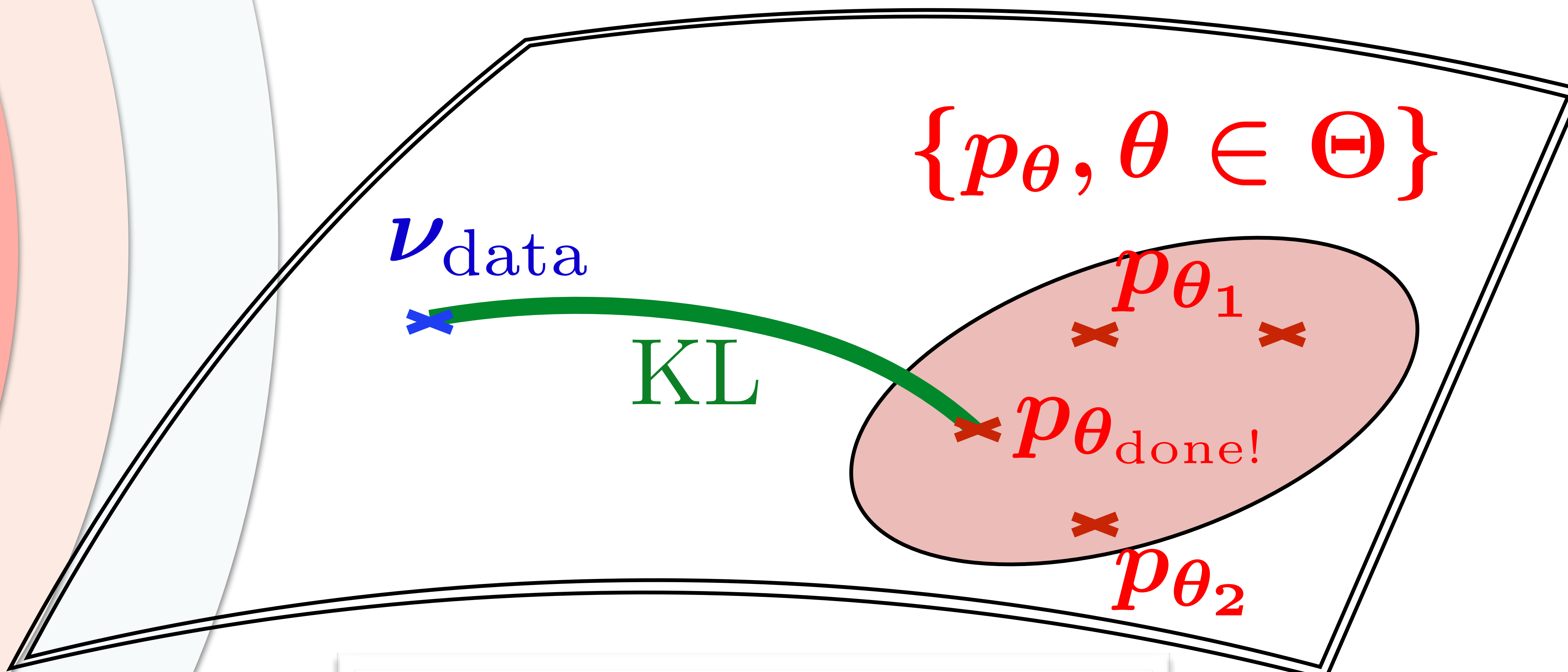
$$\log 0 = -\infty$$

$p_{\theta}(x_i)$ **must** be > 0

Maximum Likelihood Estimation

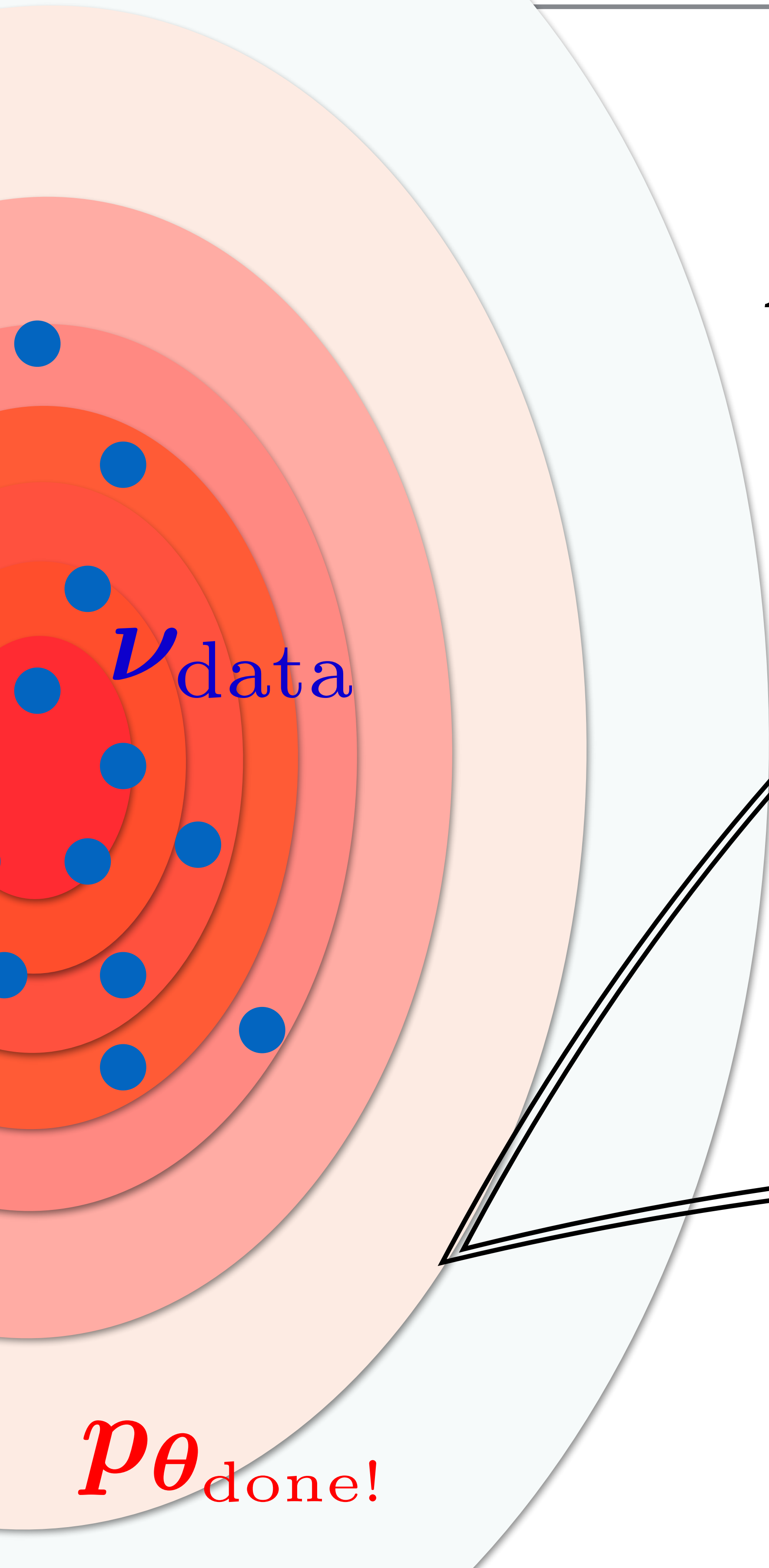
Equivalent to a KL projection in the space of probability measures



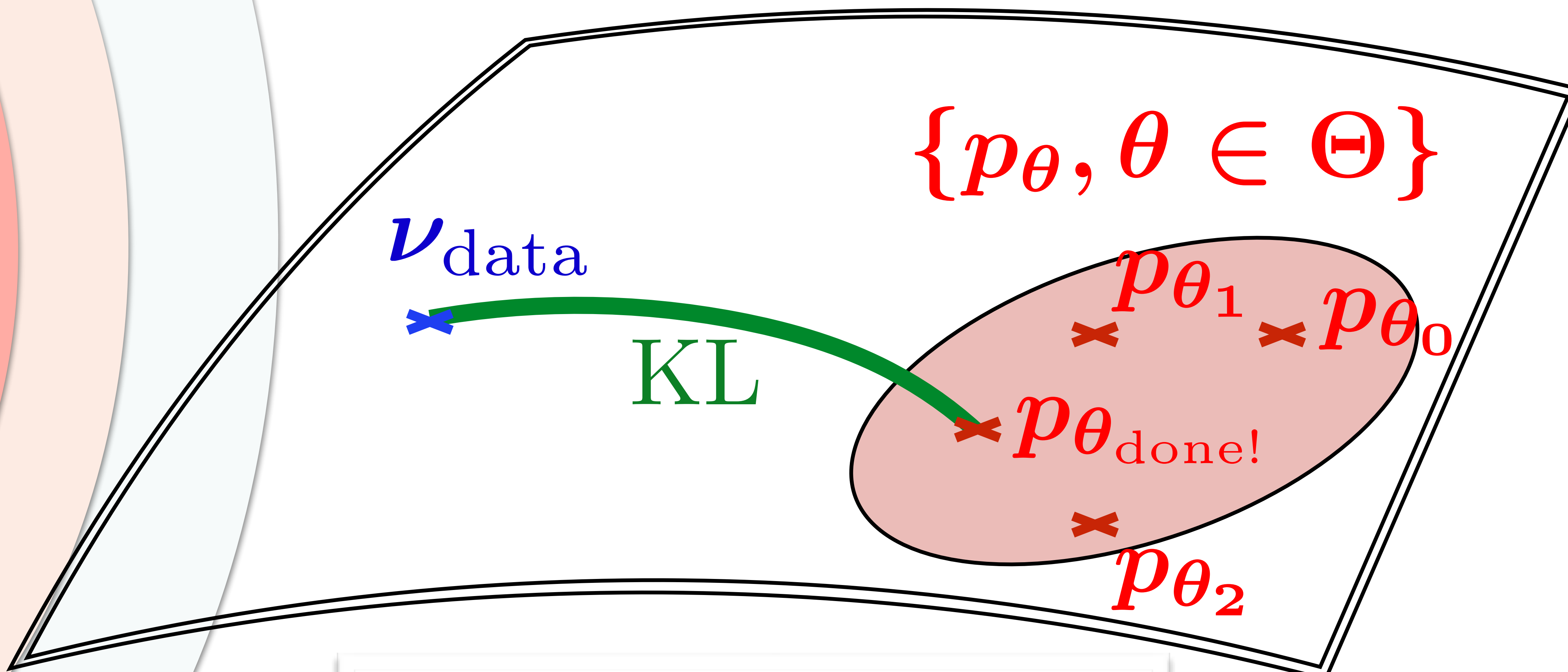


$$\min_{\theta \in \Theta} \mathrm{KL}(\nu_{\text{data}} \| p_\theta)$$

Maximum Likelihood Estimation

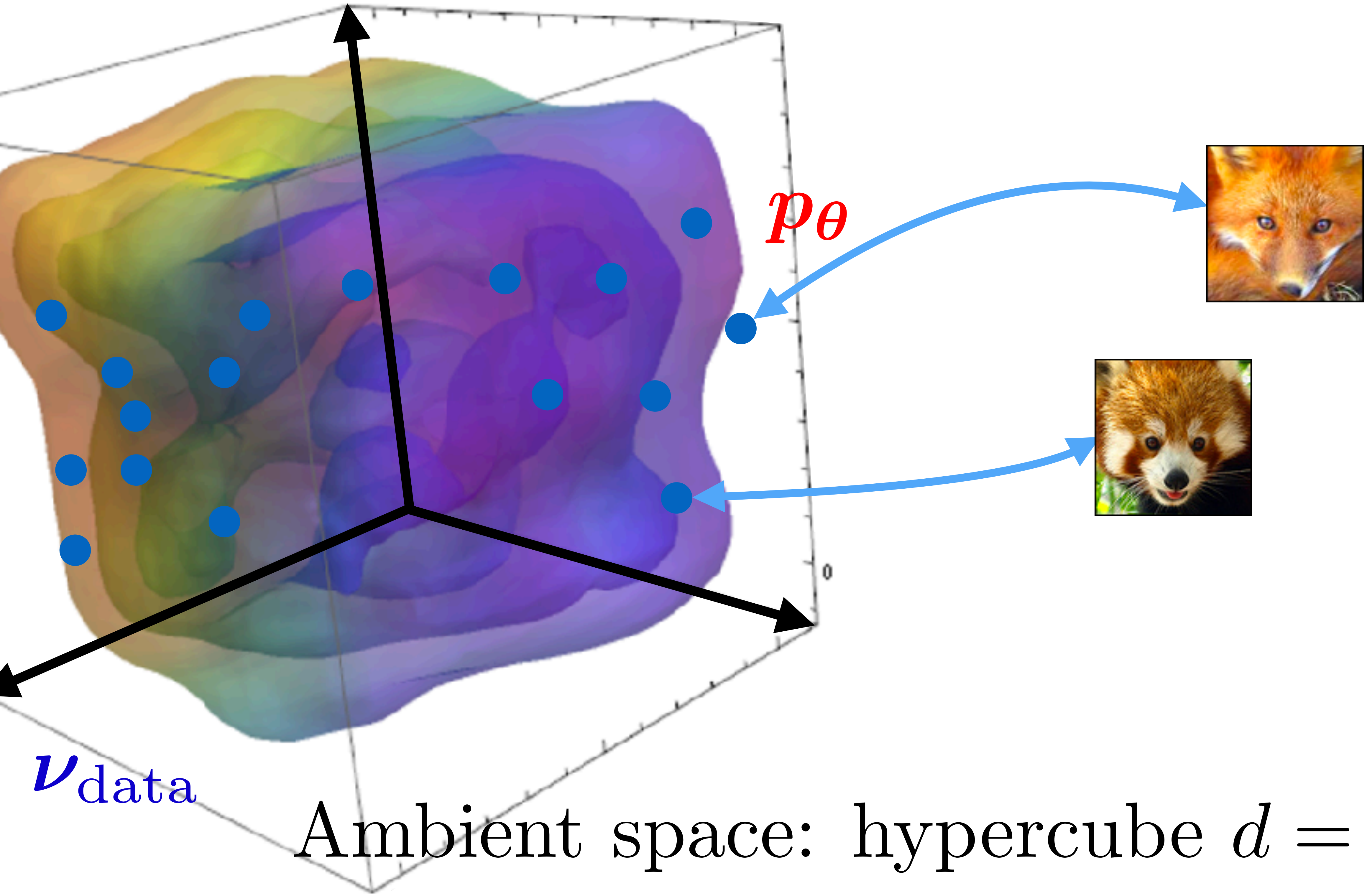


Equivalent to a KL projection in the space of probability measures



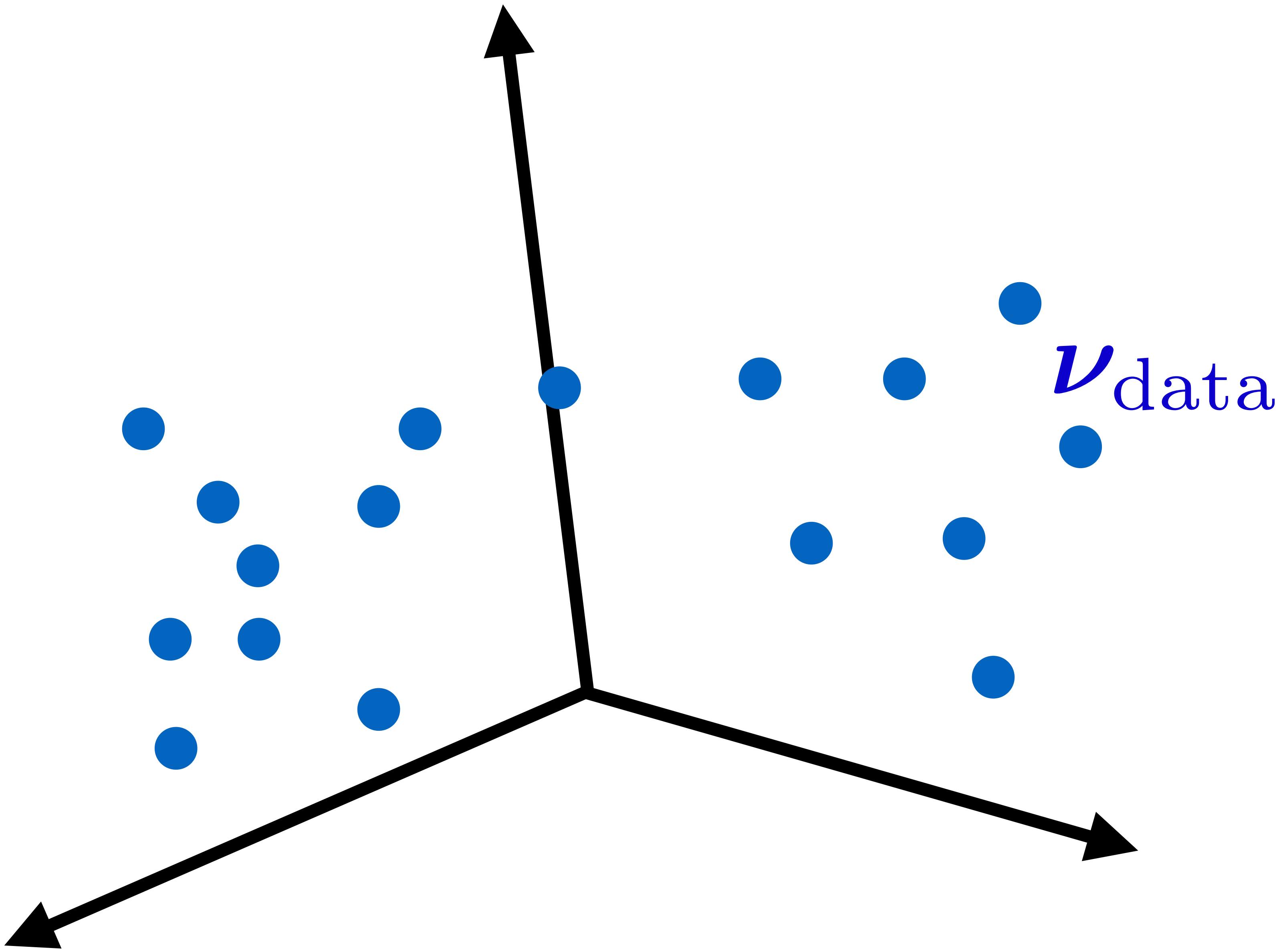
$$\min_{\boldsymbol{\theta}\in\Theta} \mathrm{KL}(\boldsymbol{\nu}_{\text{data}} \| \boldsymbol{p}_{\boldsymbol{\theta}})$$

In higher dimensional spaces...

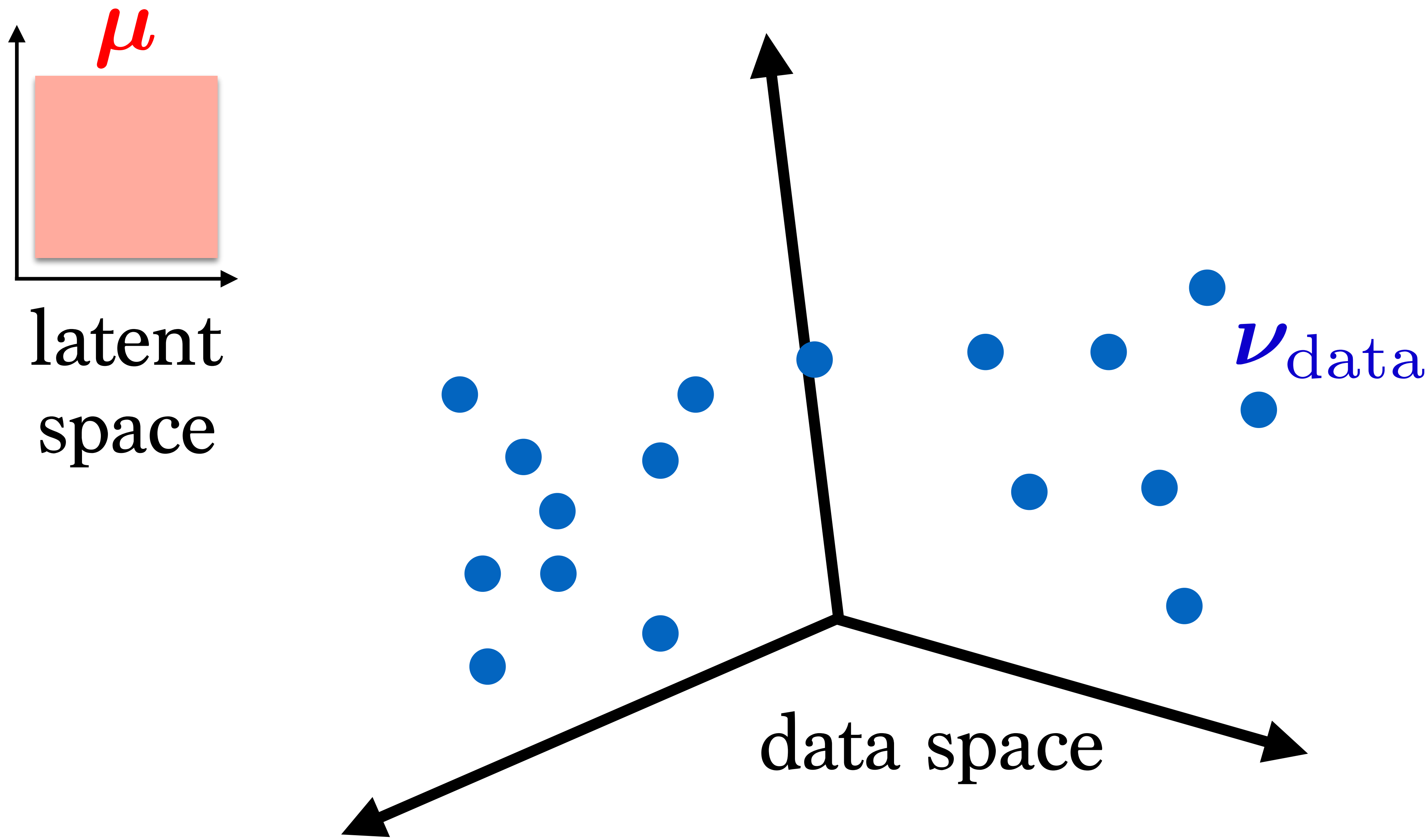


Ambient space: hypercube $d = 30.000$

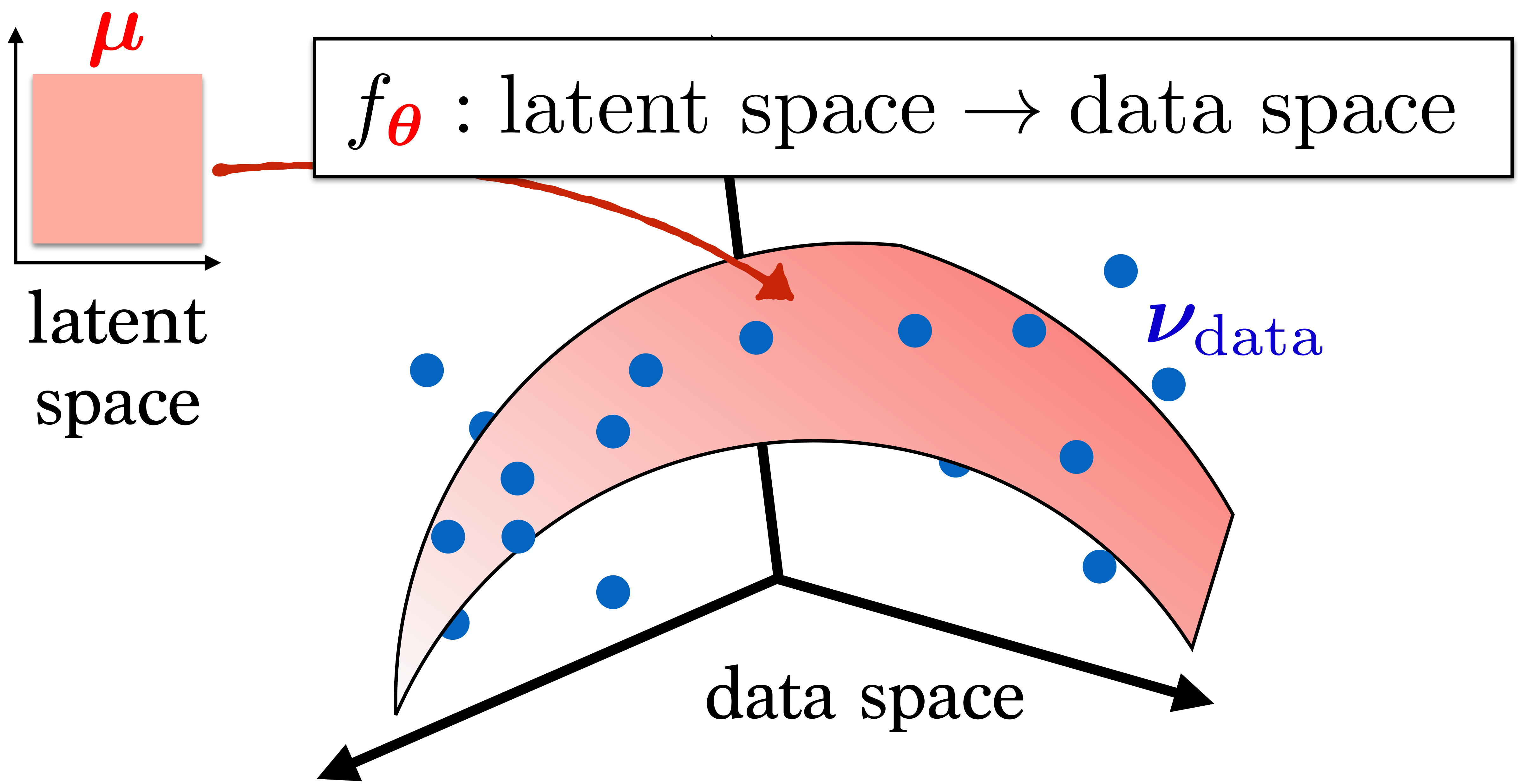
Generative Models



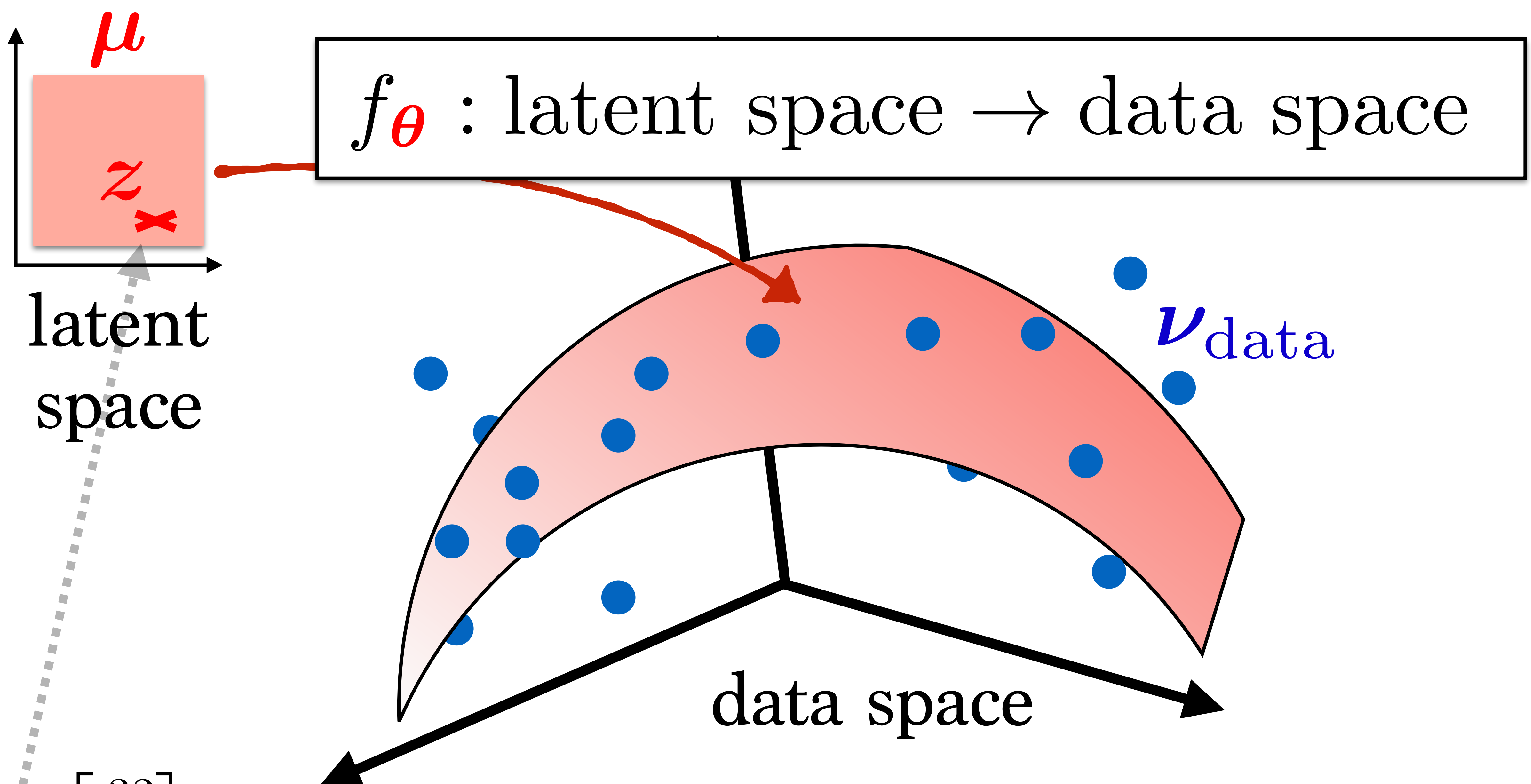
Generative Models



Generative Models

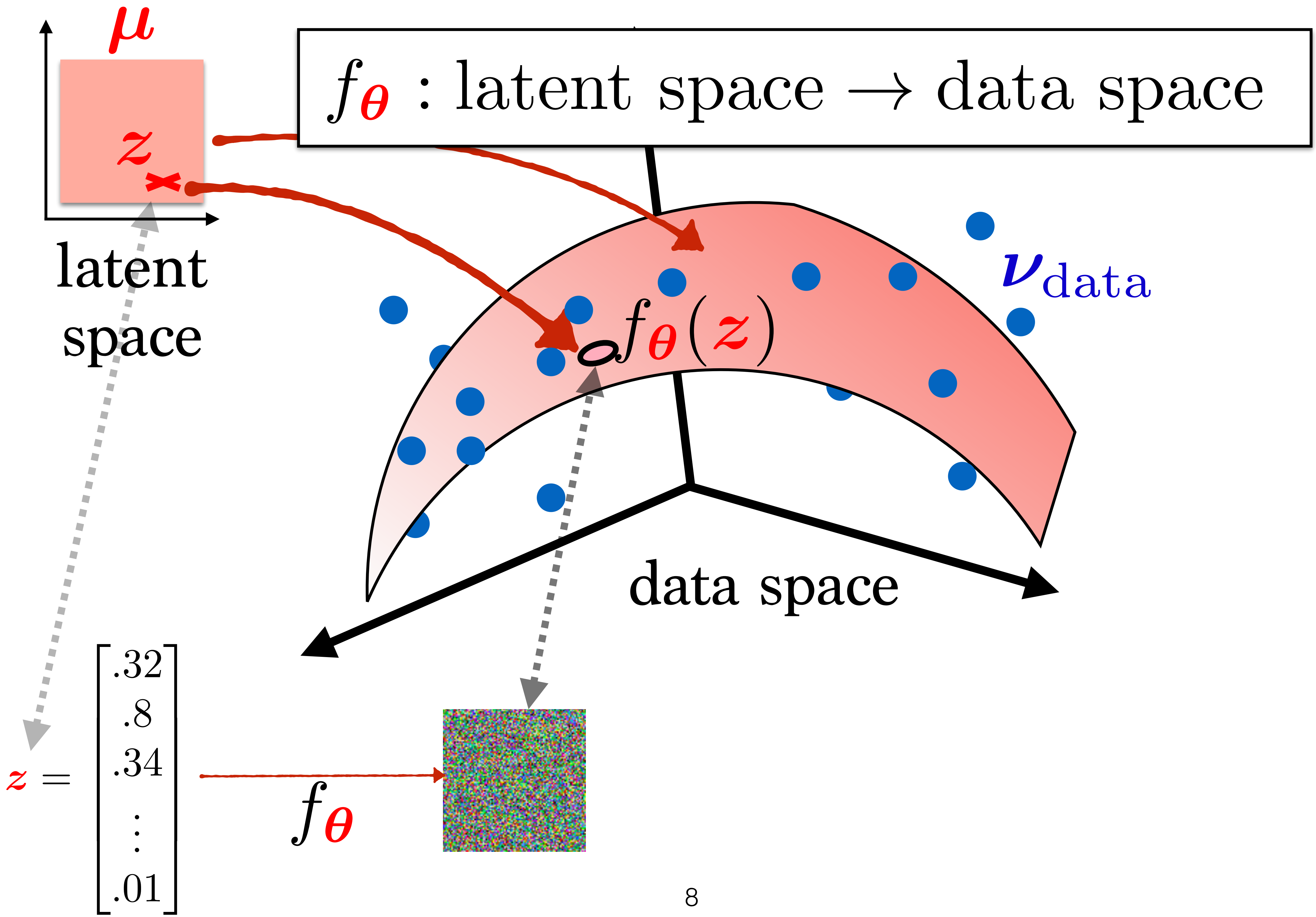


Generative Models

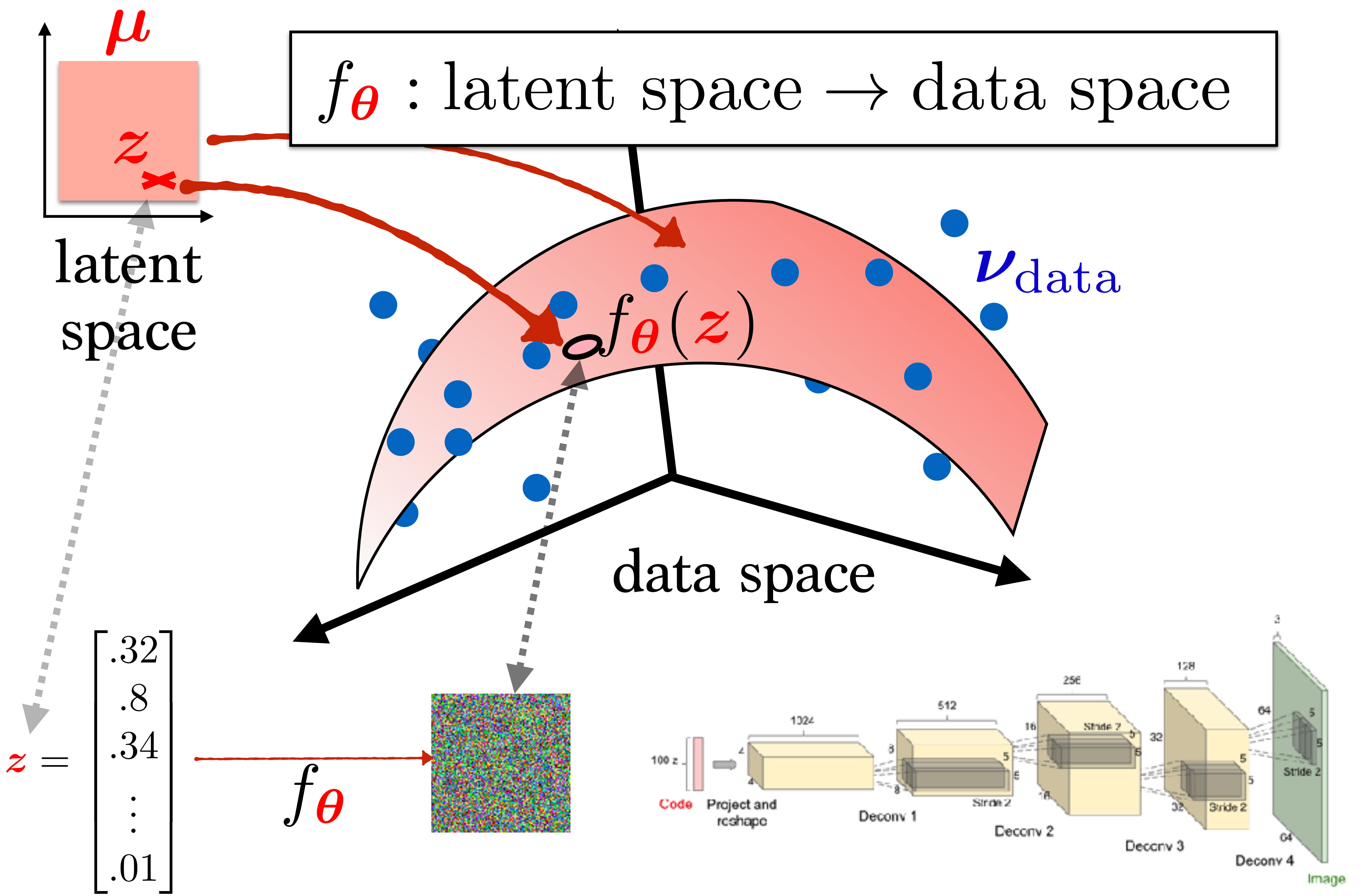


$$z = \begin{bmatrix} .32 \\ .8 \\ .34 \\ \vdots \\ .01 \end{bmatrix}$$

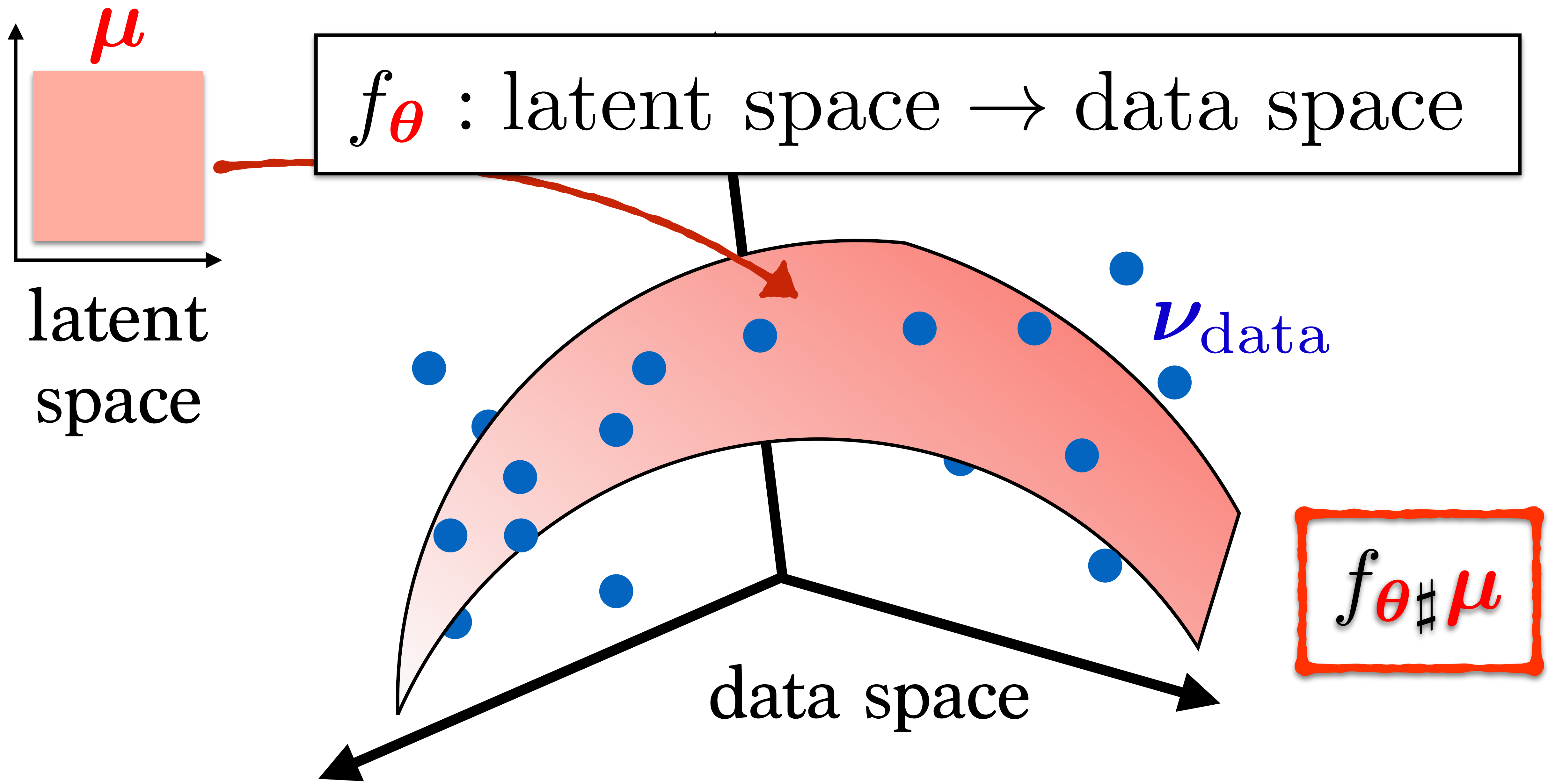
Generative Models



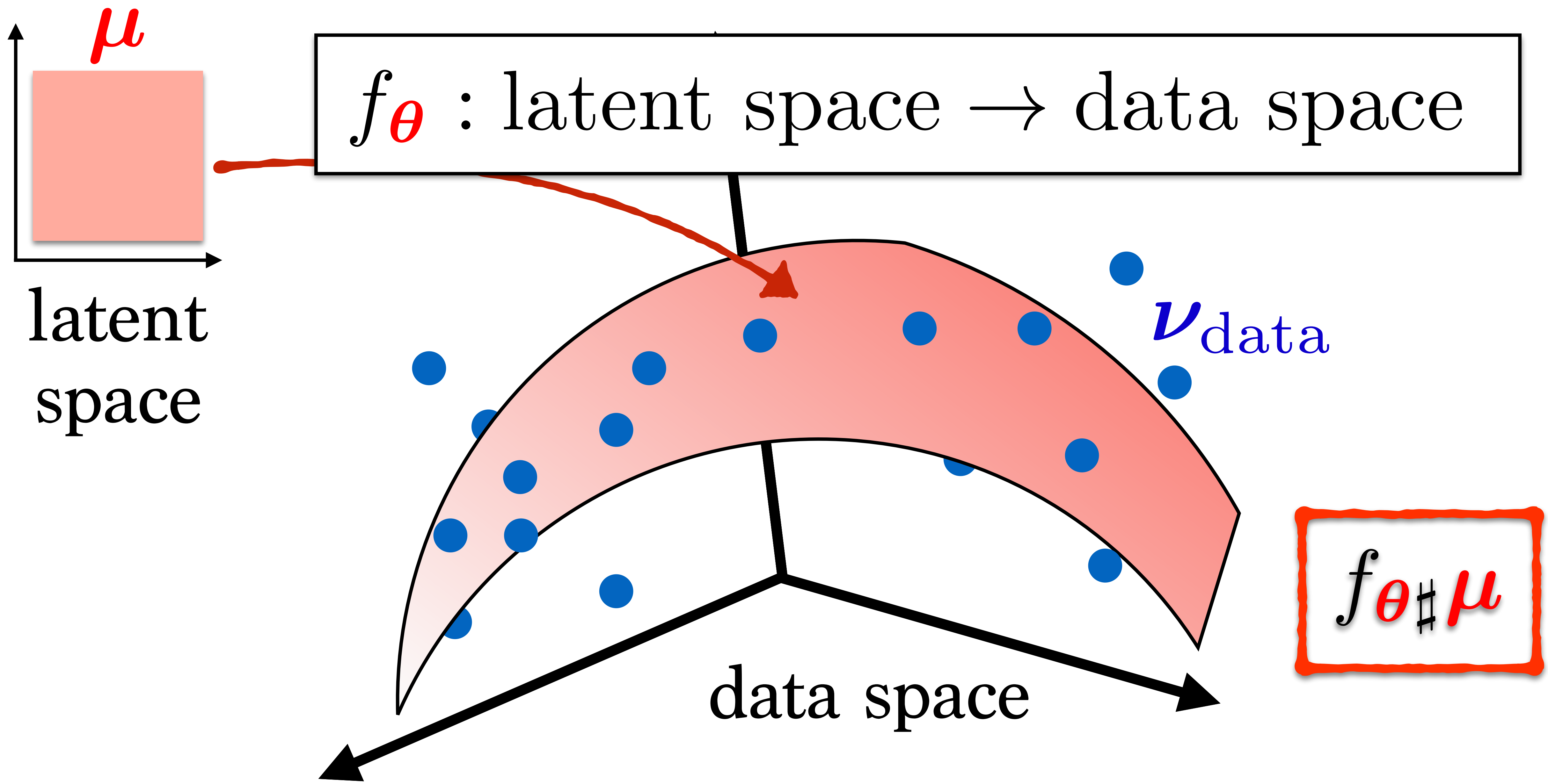
Generative Models



Generative Models

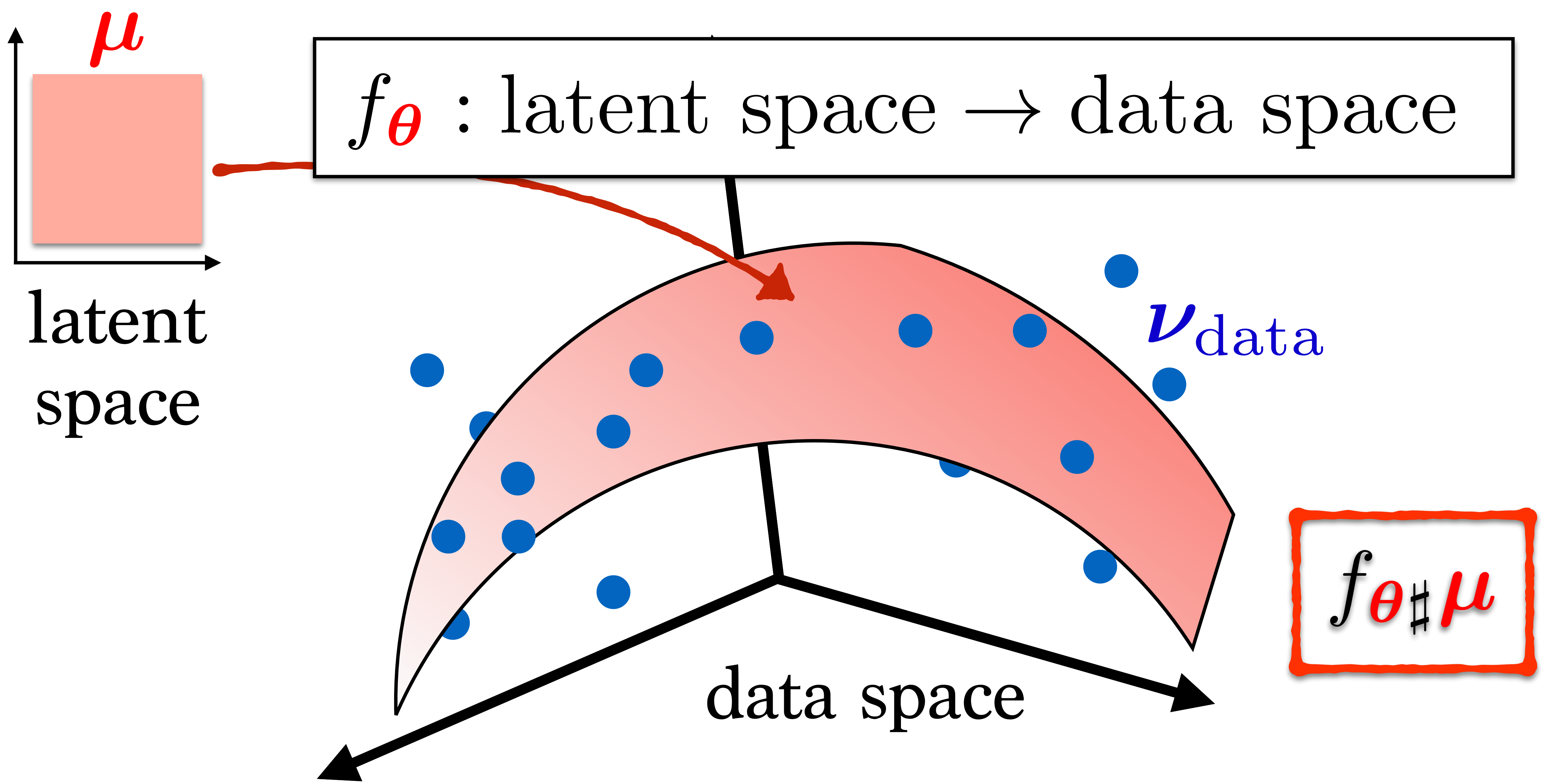


Generative Models



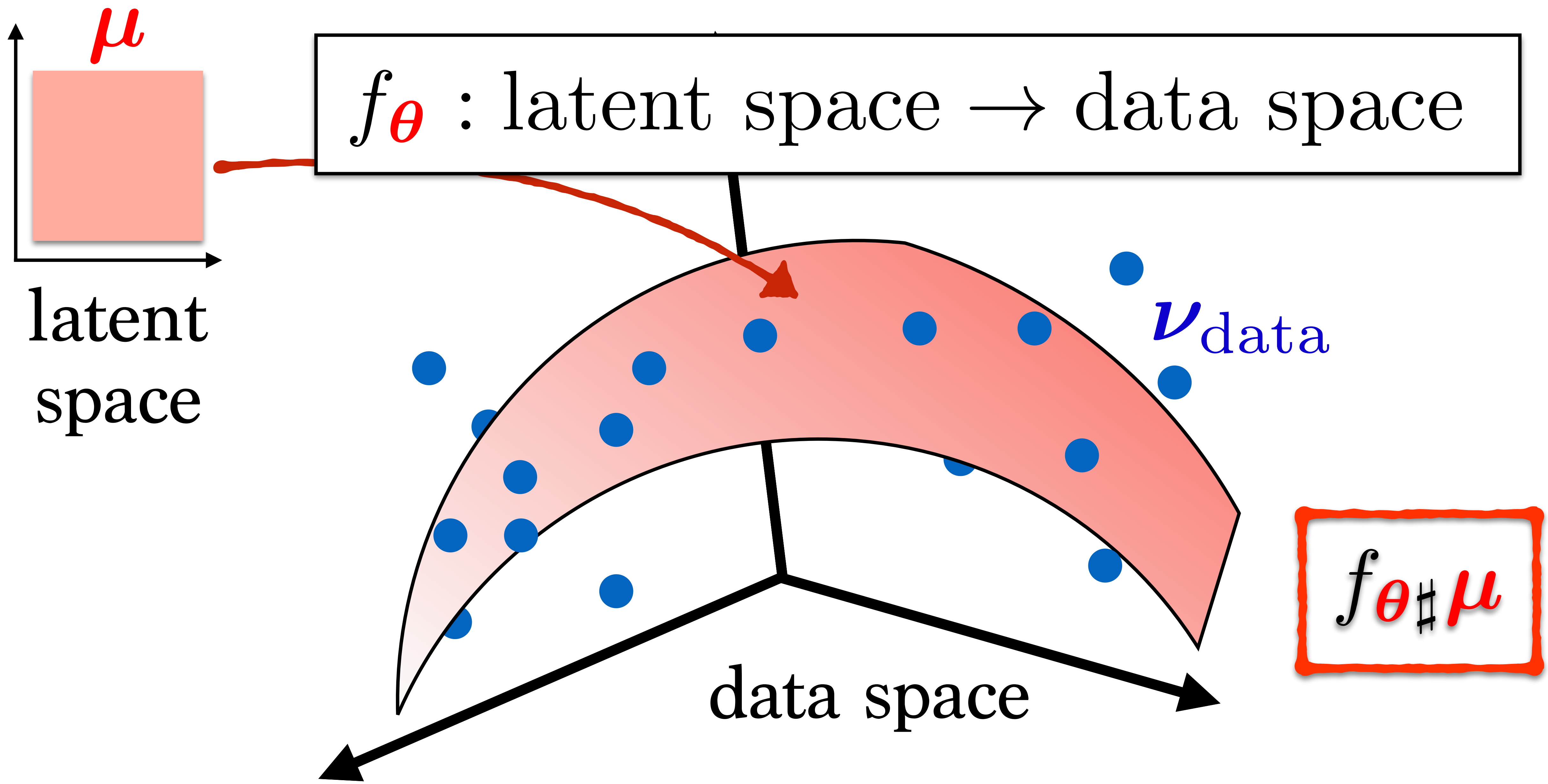
Goal: find θ such that $f_{\theta\sharp}\mu$ fits ν_{data}

Generative Models



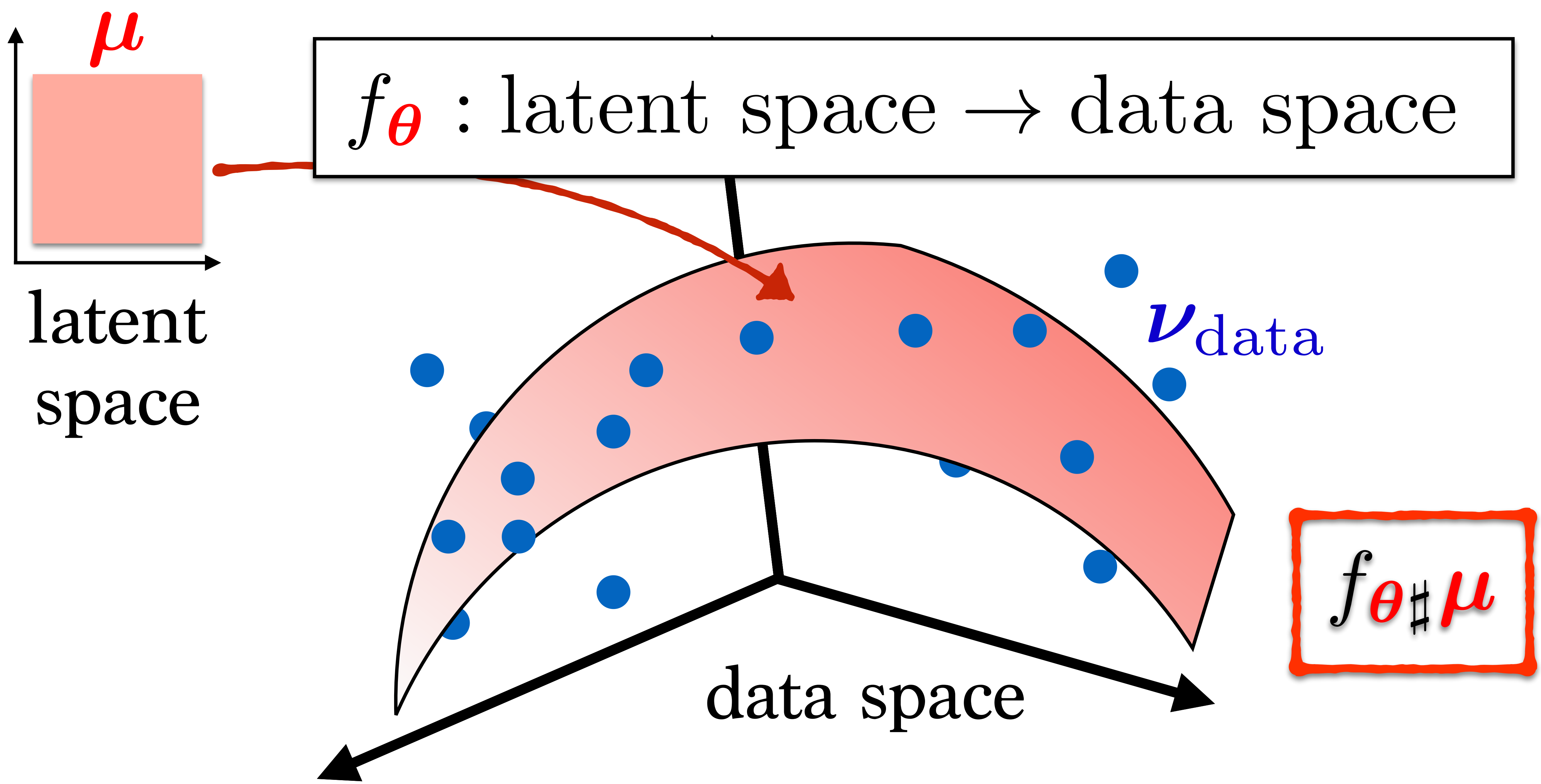
Goal: find θ such that $f_{\theta\sharp}\mu$ fits ν_{data}

Generative Models



MLE $\quad \max_{\boldsymbol{\theta}\in\Theta} \frac{1}{N}\sum_{i=1}^{N} \log \boldsymbol{p}_{\boldsymbol{\theta}}(\boldsymbol{x}_i) = \min_{\boldsymbol{\theta}\in\Theta} \mathrm{KL}(\boldsymbol{\nu}_{\text{data}} \| \boldsymbol{p}_{\boldsymbol{\theta}})$

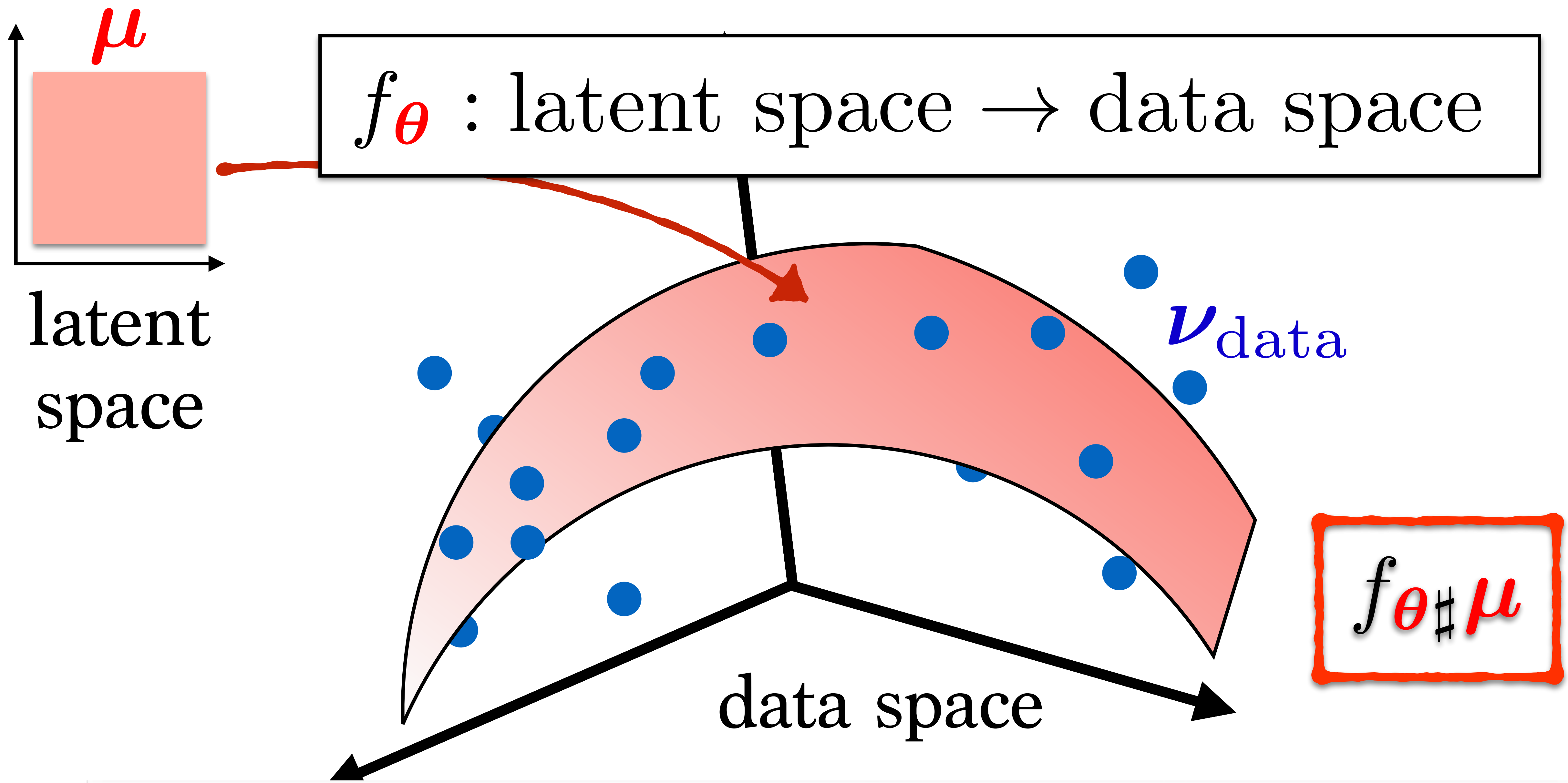
Generative Models



~~MLE~~

$$\max_{\theta\in\Theta} \frac{1}{N}\sum_{i=1}^{N} \log f_{\theta\sharp}\mu(x_i) \qquad \min_{\theta\in\Theta} \mathrm{KL}(\nu_{\text{data}} \| f_{\theta\sharp}\mu)$$

Generative Models

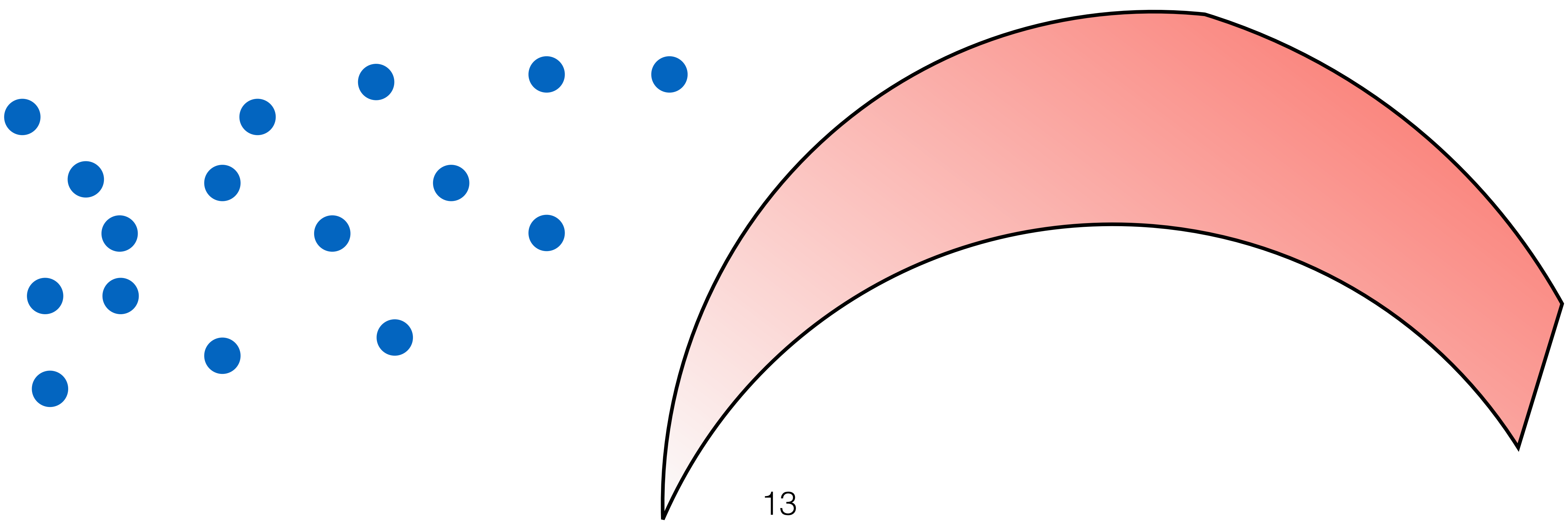


Need a more flexible **discrepancy function** to compare ν_{data} and $f_{\theta\sharp}\mu$

Workarounds?

- Formulation as **adversarial problem** **[GPM…'14]**

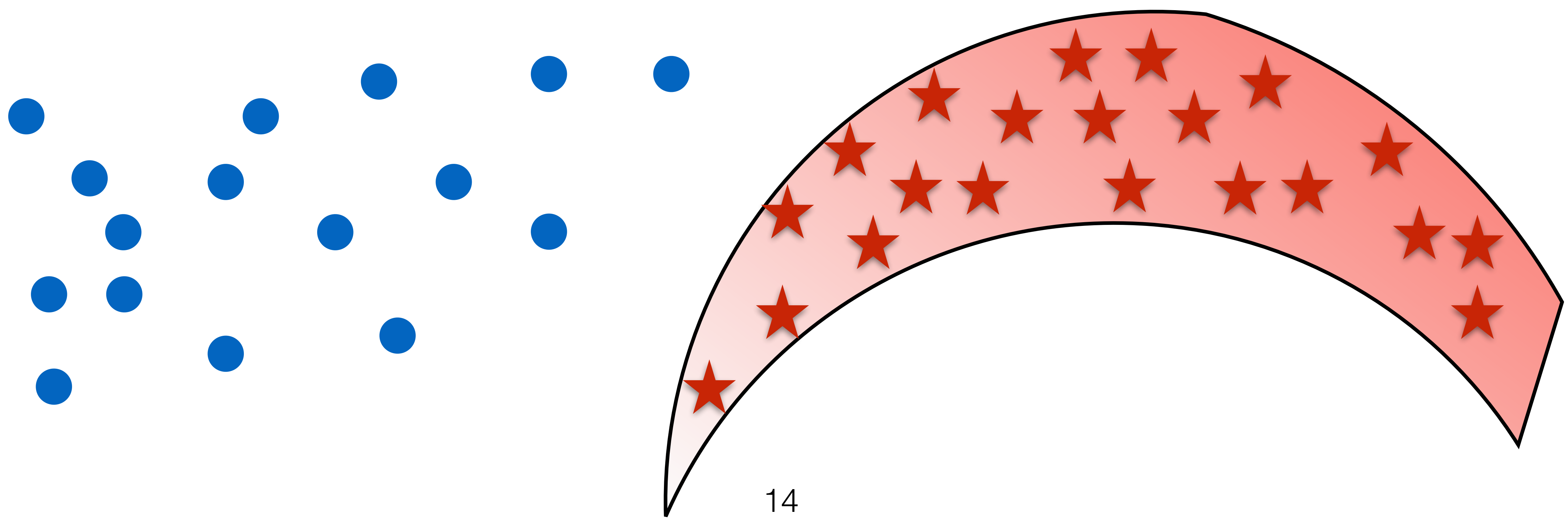
$$\min_{\theta \in \Theta} \max_{\text{classifiers } g} \text{Accuracy}_{g}\left(\left(f_{\theta\sharp}\mu, +1\right), \left(\nu_{\text{data}}, -1\right)\right)$$



Workarounds?

- Formulation as **adversarial problem** **[GPM...'14]**

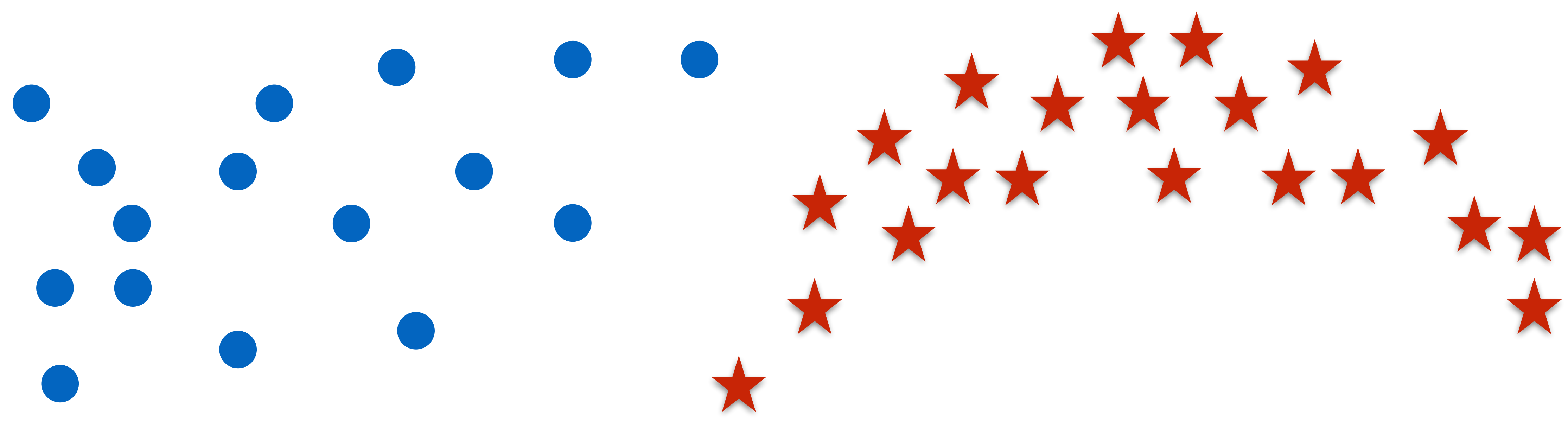
$$\min_{\theta \in \Theta} \max_{\text{classifiers } g} \text{Accuracy}_g\left(\left(f_\theta \sharp \mu, +1\right), \left(\nu_{\text{data}}, -1\right)\right)$$



Workarounds?

- Formulation as **adversarial problem** **[GPM…'14]**

$$\min_{\boldsymbol{\theta}\in\Theta}\ \max_{\text{classifiers } \boldsymbol{g}}\ \text{Accuracy}_{\boldsymbol{g}}\left(\left(\boldsymbol{f_{\theta\sharp}\mu}, +1\right), \left(\boldsymbol{\nu}_{\text{data}}, -1\right)\right)$$



Workarounds?

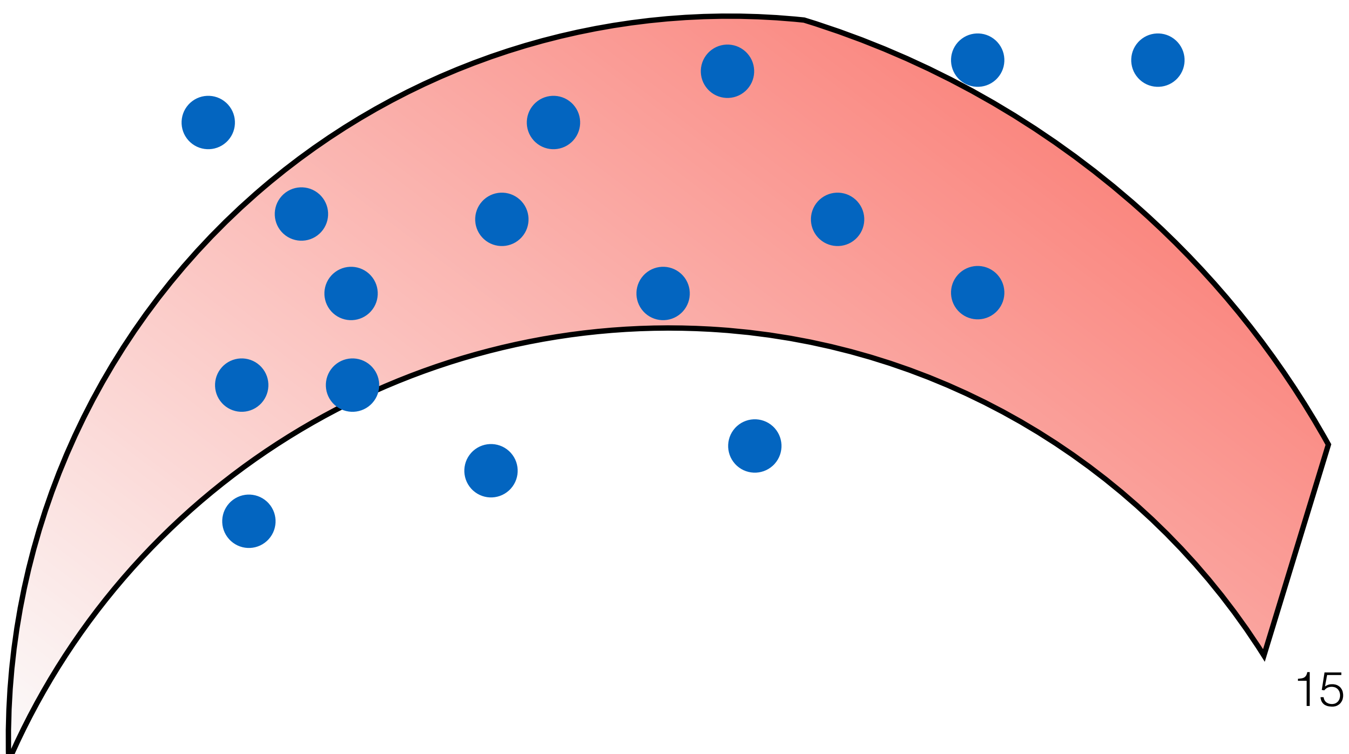
- Formulation as **adversarial problem** **[GPM…'14]**

$$\min_{\theta\in\Theta}\ \max_{\text{classifiers } g}\ \text{Accuracy}_g\left((f_{\theta\sharp}\mu, +1), (\nu_{\text{data}}, -1)\right)$$

Workarounds?

- Formulation as **adversarial problem** **[GPM…'14]**

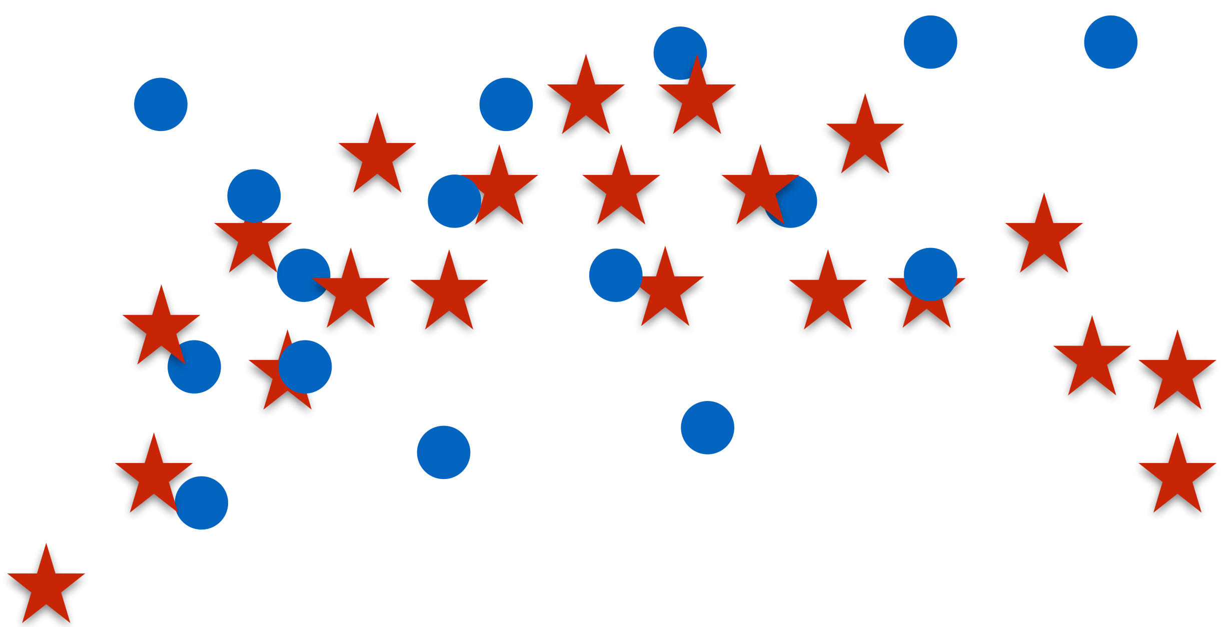
$$\min_{\theta \in \Theta} \max_{\text{classifiers } g} \text{Accuracy}_g\left(\left(f_\theta \sharp \mu, +1\right), \left(\nu_{\text{data}}, -1\right)\right)$$



Workarounds?

- Formulation as **adversarial problem** **[GPM…'14]**

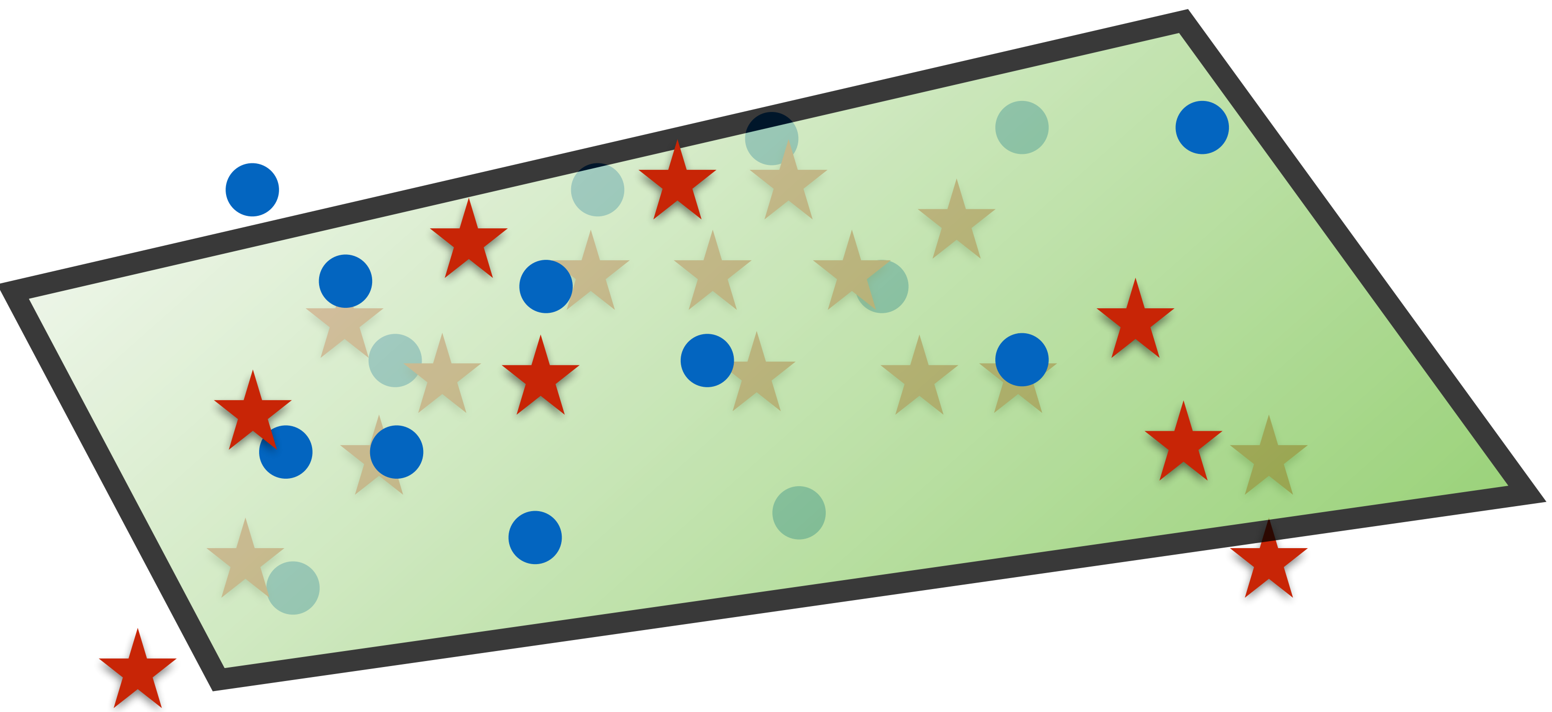
$$\min_{\theta\in\Theta} \max_{\text{classifiers } g} \text{Accuracy}_g\left(\left(f_{\theta\sharp}\mu, +1\right), \left(\nu_{\text{data}}, -1\right)\right)$$



Workarounds?

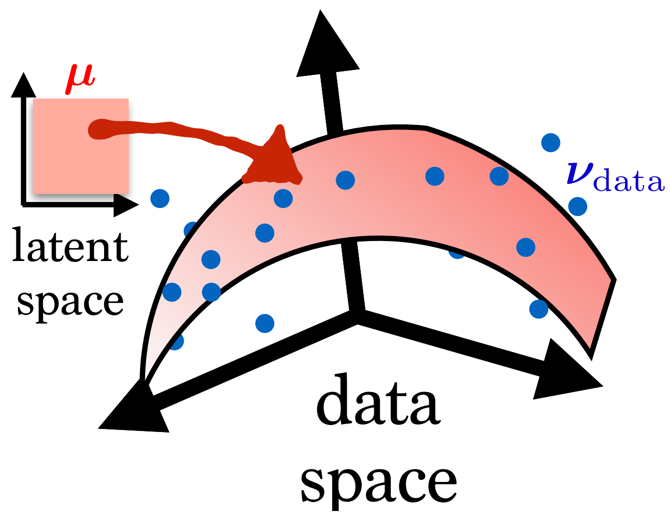
- Formulation as **adversarial problem** **[GPM…'14]**

$$\min_{\theta\in\Theta} \max_{\text{classifiers } g} \text{Accuracy}_g\left(\left(f_{\theta\sharp}\mu, +1\right), \left(\nu_{\text{data}}, -1\right)\right)$$



low classification accuracy…
is the goal.

Another idea?



- Use a **metric** Δ for probability measures, that can handle measures with non-overlapping supports:

$$\min_{\theta \in \Theta} \Delta(\nu_{\mathrm{data}}, p_{\theta}), \qquad \textbf{not} \min_{\theta \in \Theta} \mathrm{KL}(\nu_{\mathrm{data}} \| p_{\theta})$$

Minimum △ Estimation

The Annals of Statistics
1980, Vol. 8, No. 3, 457–487

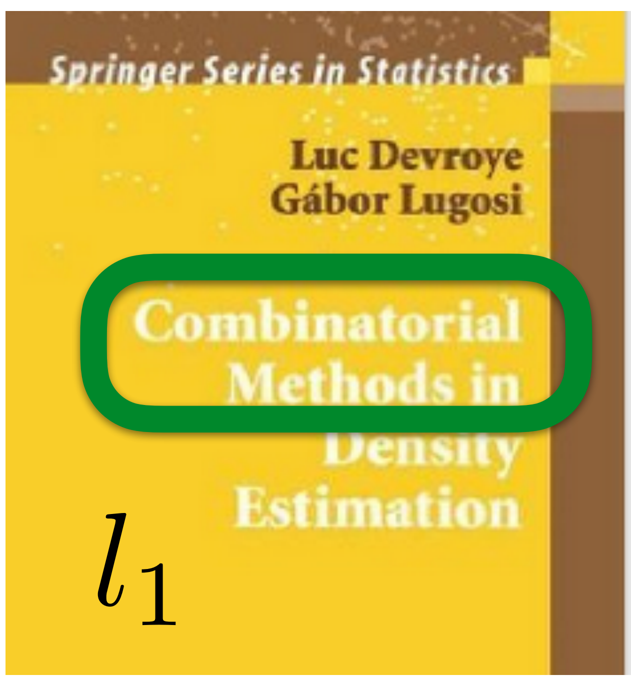
MINIMUM CHI-SQUARE, NOT MAXIMUM LIKELIHOOD!

BY JOSEPH BERKSON

Mayo Clinic, Rochester, Minnesota

ELSEVIER Computational Statistics & Data Analysis 29 (1998) 81–103

COMPUTATIONAL STATISTICS & DATA ANALYSIS



Minimum Hellinger distance estimation for Poisson mixtures

Dimitris Karlis, Evdokia Xekalaki*

Department of Statistics, Athens University of Economics and Business, 76 Patission Str., 104 34 Athens, Greece

ELSEVIER

Available online at www.sciencedirect.com

SCIENCE DIRECT

Statistics & Probability Letters 76 (2006) 1298–1302

STATISTICS & PROBABILITY LETTERS

www.elsevier.com/locate/stapro

On minimum Kantorovich distance estimators

Federico Bassetti[a], Antonella Bodini[b], Eugenio Regazzini[a,*]

△ Generative Model Estimation

Generative Moment Matching Networks

Yujia Li[1] YUJIALI@CS.TORONTO.EDU
Kevin Swersky[1] KSWERSKY@CS.TORONTO.EDU
Richard Zemel[1,2] ZEMEL@CS.TORONTO.EDU
[1]Department of Computer Science, University of Toronto, Toronto, ON, CANADA
[2]Canadian Institute for Advanced Research, Toronto, ON, CANADA

MMD GAN: Towards Deeper Understanding of Moment Matching Network

Chun-Liang Li[1,*] **Wei-Cheng Chang**[1,*] **Yu Cheng**[2] **Yiming Yang**[1] **Barnabás Póczos**[1]
[1] Carnegie Mellon University, [2]IBM Research
{chunlial,wchang2,yiming,bapoczos}@cs.cmu.edu chengyu@us.ibm.com

Training generative neural networks via Maximum Mean Discrepancy optimization

Gintare Karolina Dziugaite
University of Cambridge

Daniel M. Roy
University of Toronto

Zoubin Ghahramani
University of Cambridge

△ Generative Model Estimation

Generative Moment Matching Networks

Yujia Li[1] YUJIALI@CS.TORONTO.EDU
Kevin Swersky[1] KSWERSKY@CS.TORONTO.EDU
Richard Zemel[1,2] ZEMEL@CS.TORONTO.EDU
[1]Department of Computer Science, University of Toronto, Toronto, ON, CANADA
[2]Canadian Institute for Advanced Research, Toronto, ON, CANADA

Training generative neural networks via Maximum Mean Discrepancy optimization

Gintare Karolina Dziugaite University of Cambridge
Daniel M. Roy University of Toronto
Zoubin Ghahramani University of Cambridge

MMD GAN: Towards Deeper Understanding of Moment Matching Network

Chun-Liang Li[1,*] **Wei-Cheng Chang**[1,*] **Yu Cheng**[2] **Yiming Yang**[1] **Barnabás Póczos**[1]
[1] Carnegie Mellon University, [2]IBM Research
{chunlial,wchang2,yiming,bapoczos}@cs.cmu.edu chengyu@us.ibm.com

Wasserstein Training of Restricted Boltzmann Machines

Grégoire Montavon Technische Universität Berlin gregoire.montavon@tu-berlin.de
Klaus-Robert Müller* Technische Universität Berlin klaus-robert.mueller@tu-berlin.de
Marco Cuturi CREST, ENSAE, Université Paris-Saclay marco.cuturi@ensae.fr

Inference in generative models using the Wasserstein distance

Espen Bernton, Mathieu Gerber, Pierre E. Jacob, Christian P. Robert

Wasserstein GAN

Martin Arjovsky[1], Soumith Chintala[2], and Léon Bottou[1,2]
[1]Courant Institute of Mathematical Sciences
[2]Facebook AI Research

△ Generative Model Estimation

Generative Moment Matching Networks

Yujia Li[1] YUJIALI@CS.TORONTO.EDU
Kevin Swersky[1] KSWERSKY@CS.TORONTO.EDU
Richard Zemel[1,2] ZEMEL@CS.TORONTO.EDU
[1]Department of Computer Science, University of Toronto, Toronto, ON, CANADA
[2]Canadian Institute for Advanced Research, Toronto, ON, CANADA

Training generative neural networks via Maximum Mean Discrepancy optimization

Gintare Karolina Dziugaite University of Cambridge
Daniel M. Roy University of Toronto
Zoubin Ghahramani University of Cambridge

MMD GAN: Towards Deeper Understanding of Moment Matching Network

Chun-Liang Li[1,*] **Wei-Cheng Chang**[1,*] **Yu Cheng**[2] **Yiming Yang**[1] **Barnabás Póczos**[1]
[1] Carnegie Mellon University, [2]IBM Research
{chunlial,wchang2,yiming,bapoczos}@cs.cmu.edu chengyu@us.ibm.com

Wasserstein Training of Restricted Boltzmann Machines

Grégoire Montavon Technische Universität Berlin gregoire.montavon@tu-berlin.de
Klaus-Robert Müller[*] Technische Universität Berlin klaus-robert.mueller@tu-berlin.de
Marco Cuturi CREST, ENSAE, Université Paris-Saclay marco.cuturi@ensae.fr

Inference in generative models using the Wasserstein distance

Espen Bernton, Mathieu Gerber, Pierre E. Jacob, Christian P. Robert

Wasserstein GAN

Martin Arjovsky[1], Soumith Chintala[2], and Léon Bottou[1,2]
[1]Courant Institute of Mathematical Sciences
[2]Facebook AI Research

Learning Generative Models with Sinkhorn Divergences

Aude Genevay CEREMADE, Université Paris-Dauphine
Gabriel Peyré CNRS and DMA, École Normale Supérieure
Marco Cuturi ENSAE CREST Université Paris-Saclay

Improving GANs Using Optimal Transport

Tim Salimans[*] OpenAI tim@openai.com
Han Zhang[†] Rutgers University han.zhang@cs.rutgers.edu
Alec Radford OpenAI alec@openai.com
Dimitris Metaxas Rutgers University dnm@cs.rutgers.edu

Minimum Kantorovich Estimation

- Optimal transport theory! namely *Wasserstein distances* to define discrepancy Δ.

$$\min_{\theta \in \Theta} W(\nu_{\text{data}}, f_{\theta\sharp}\mu)$$

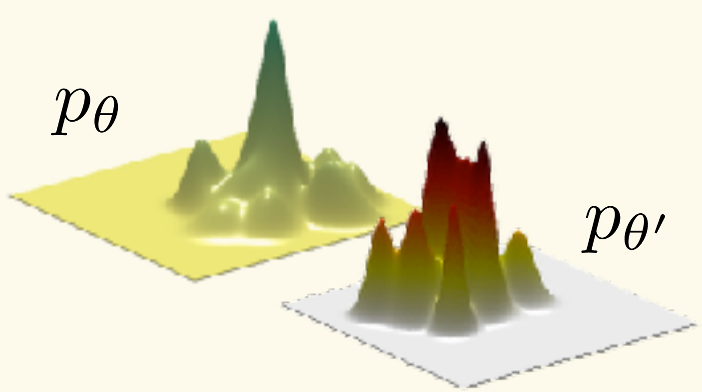
- Building on the shoulders of mathematicians...

Monge Kantorovich Koopmans Dantzig Brenier Gangbo Otto McCann Villani Figalli

Nobel'75 Fields'10 Fields'18

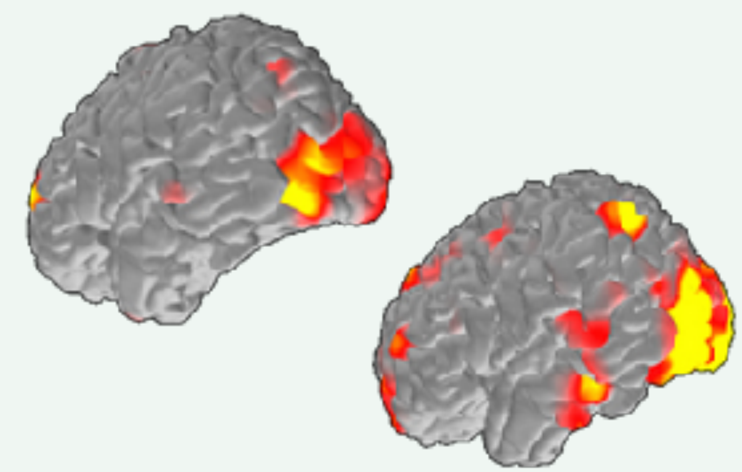
Optimal Transport in Data Sciences

The natural geometry for probability measures



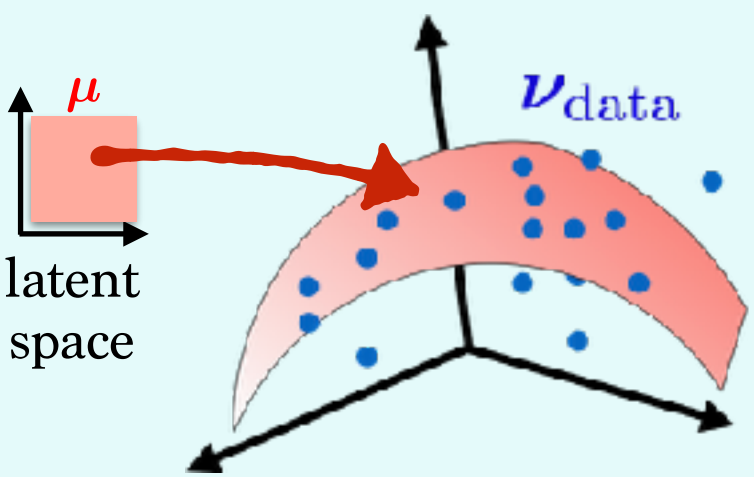
Statistical Models

Bags of features



Brain Activation Maps

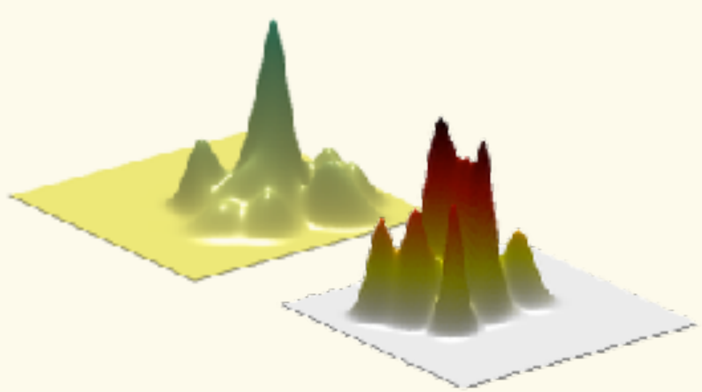
Generative Models vs. data



Color Histograms

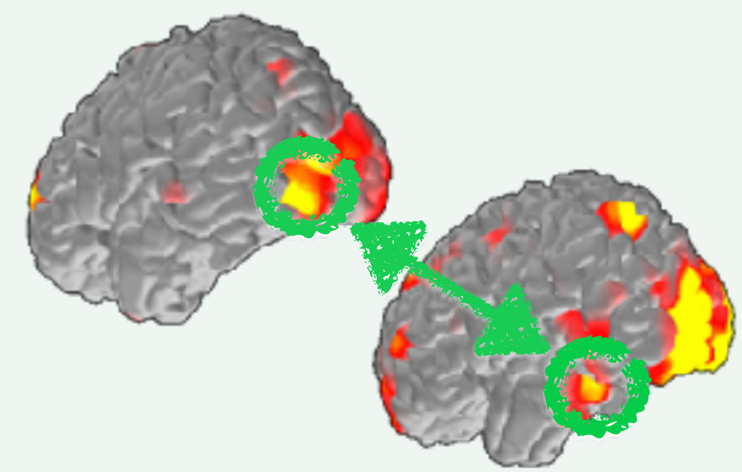
Optimal Transport in Data Sciences

The natural geometry for probability measures supported on a geometric space.



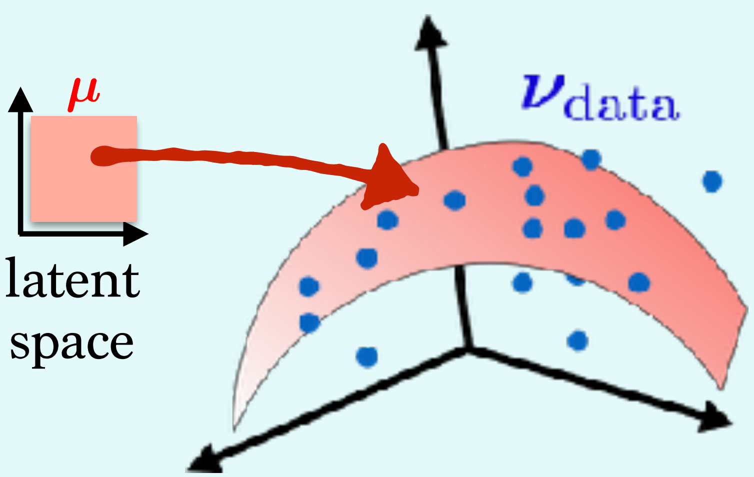
Statistical Models

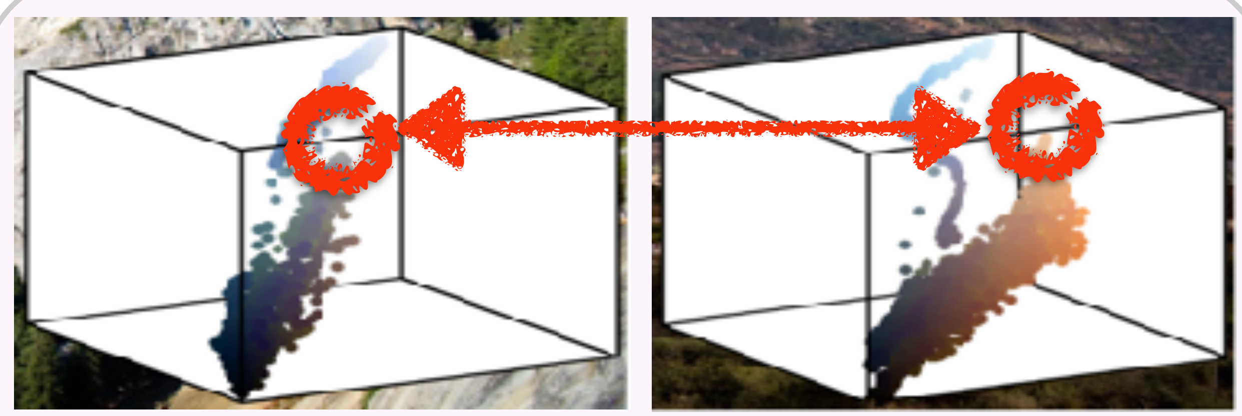
Bags of Features



Brain Activation Maps

Generative Models vs. Data

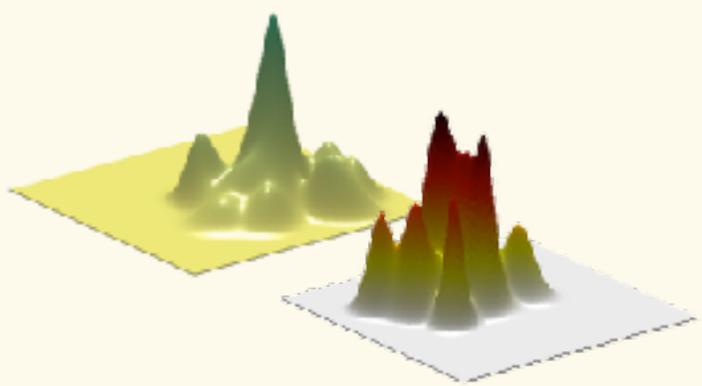




Color Histograms

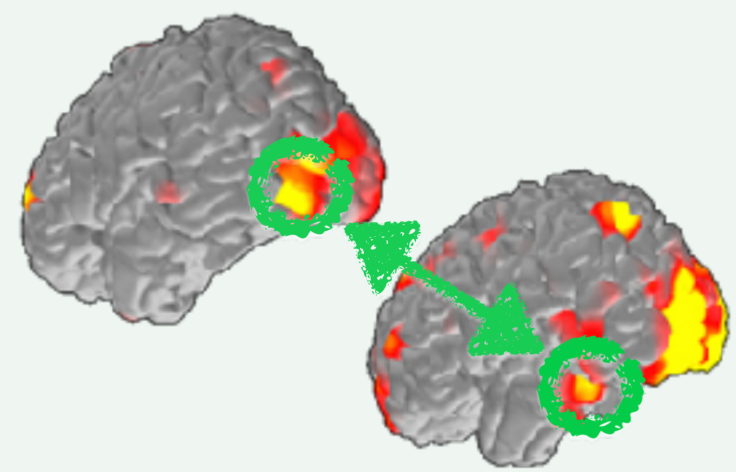
Optimal Transport in Data Sciences

The natural geometry for probability measures supported on a geometric space.



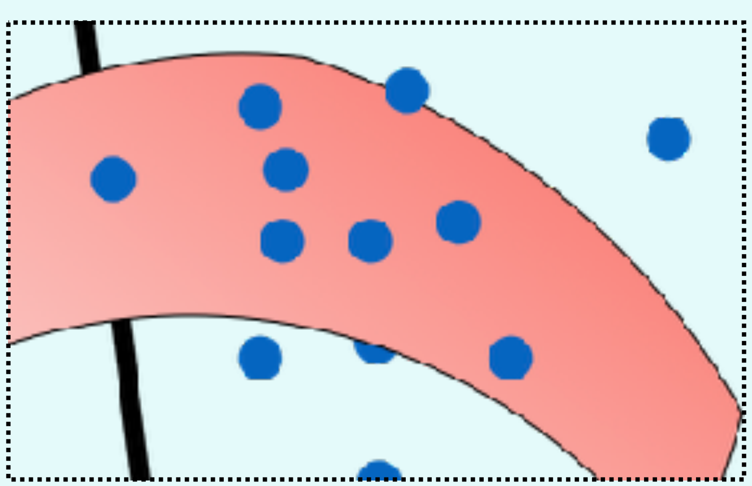
Statistical Models

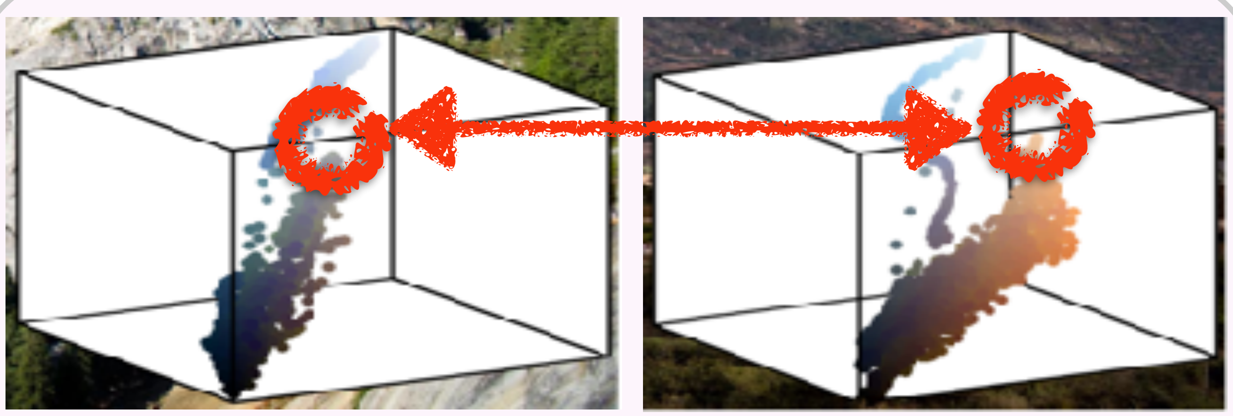
Bags of Features



Brain Activation Maps

Generative Models vs. Data

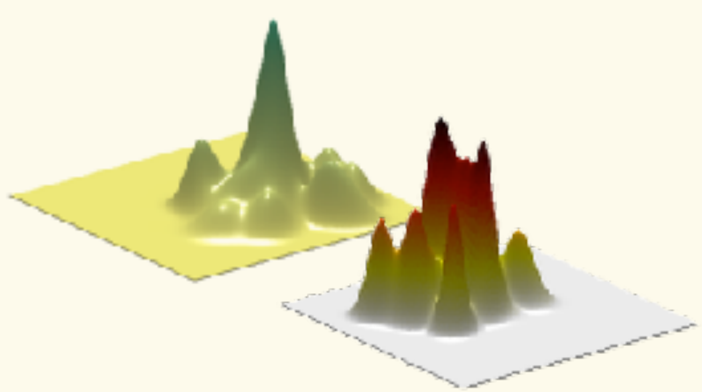




Color Histograms

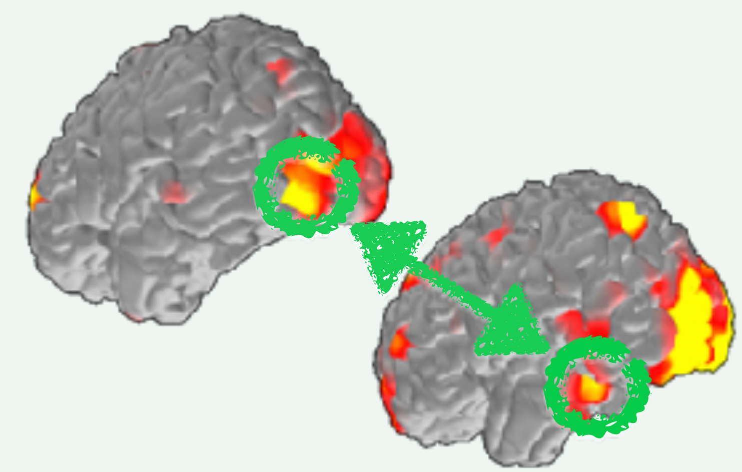
Optimal Transport in Data Sciences

The natural geometry for probability measures supported on a geometric space.



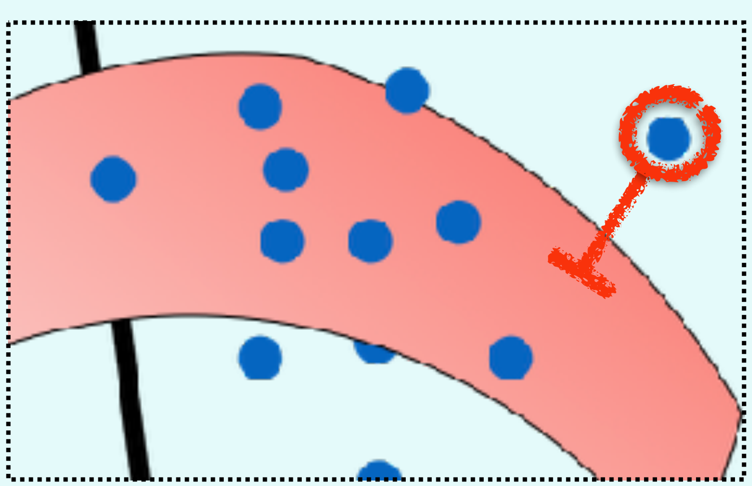
Statistical Models

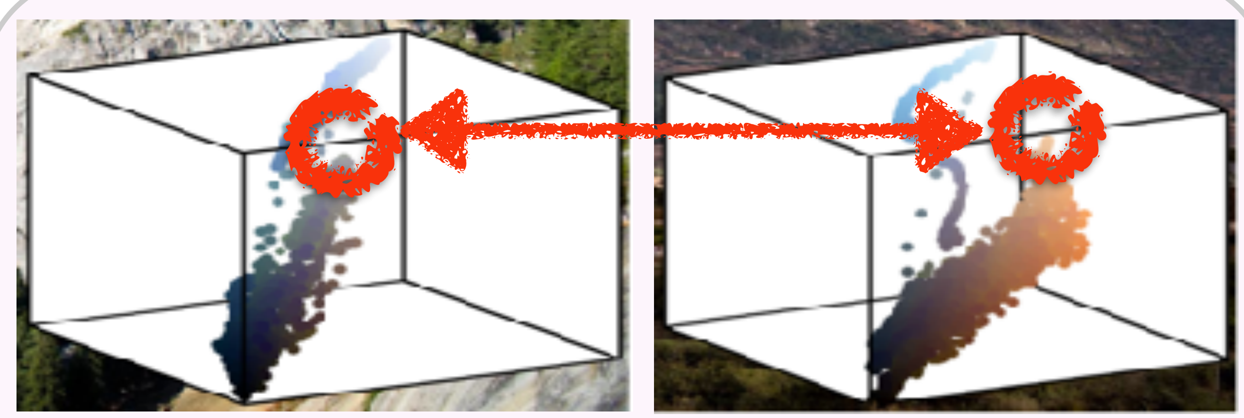
Bags of Features



Brain Activation Maps

Generative Models vs. Data





Color Histograms

Short Course Outline

1. Introduction to optimal transport
2. Computing OT exactly
3. Computing OT for data sciences
4. Some Applications

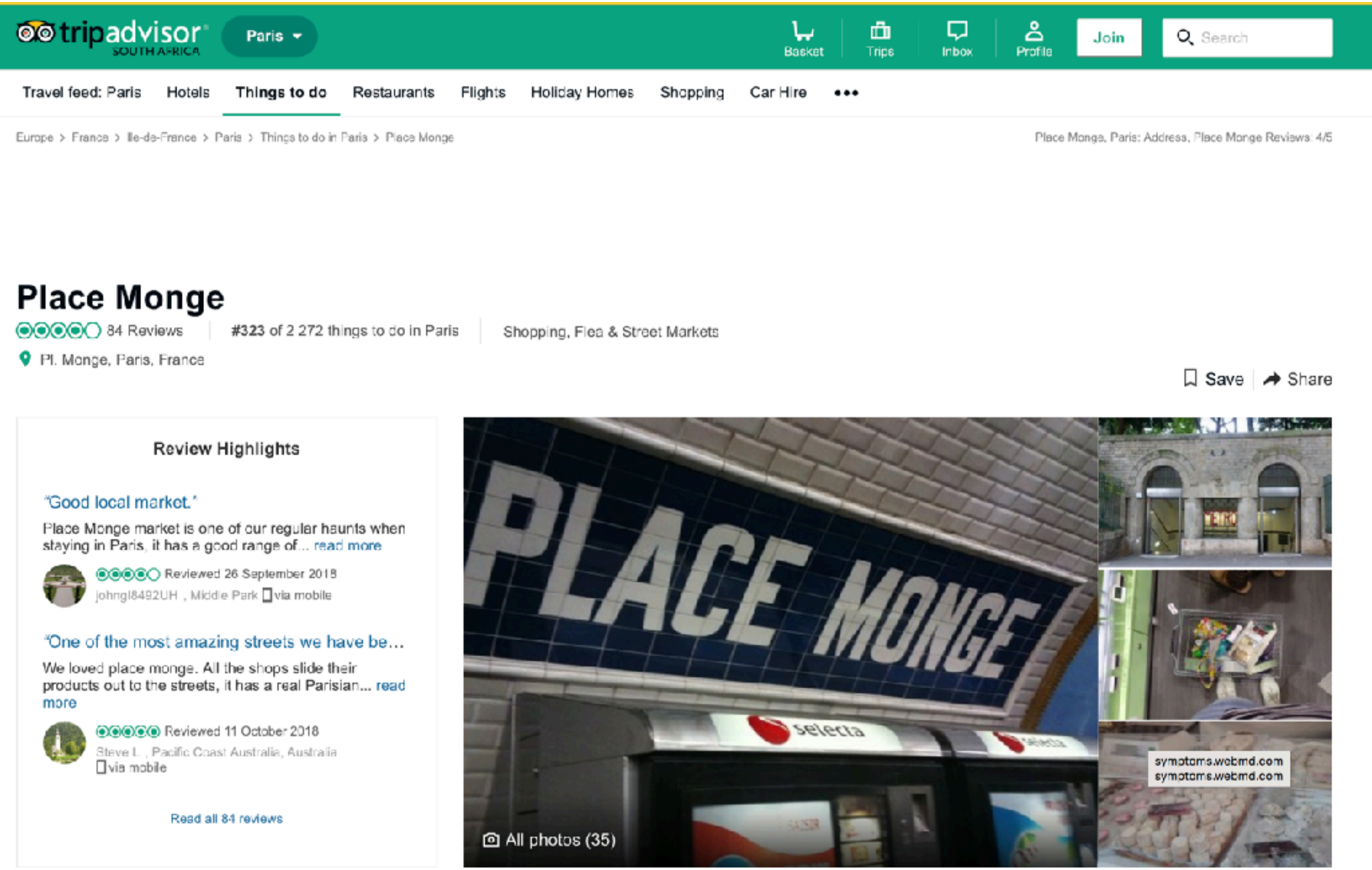
Introduction to OT

- Two examples: moving earth & soldiers
- Monge problem, Kantorovich problem
- OT as geometry, OT as a loss function

Origins: Monge Problem (1781)

Gaspard Monge (1746 - 1818)

Origins: Monge Problem (1781)



Origins: Monge Problem (1781)

MÉMOIRE SUR LA THÉORIE DES DÉBLAIS ET DES REMBLAIS.

Par M. MONGE.

LORSQU'ON doit tranſporter des terres d'un lieu dans un autre, on a coutume de donner le nom de *Déblai* au volume des terres que l'on doit tranſporter, & le nom de *Remblai* à l'eſpace qu'elles doivent occuper après le tranſport.

Origins: Monge Problem (1781)

6[illegible] MÉMOIRES DE L'ACADÉMIE ROYALE

MÉMOIRE

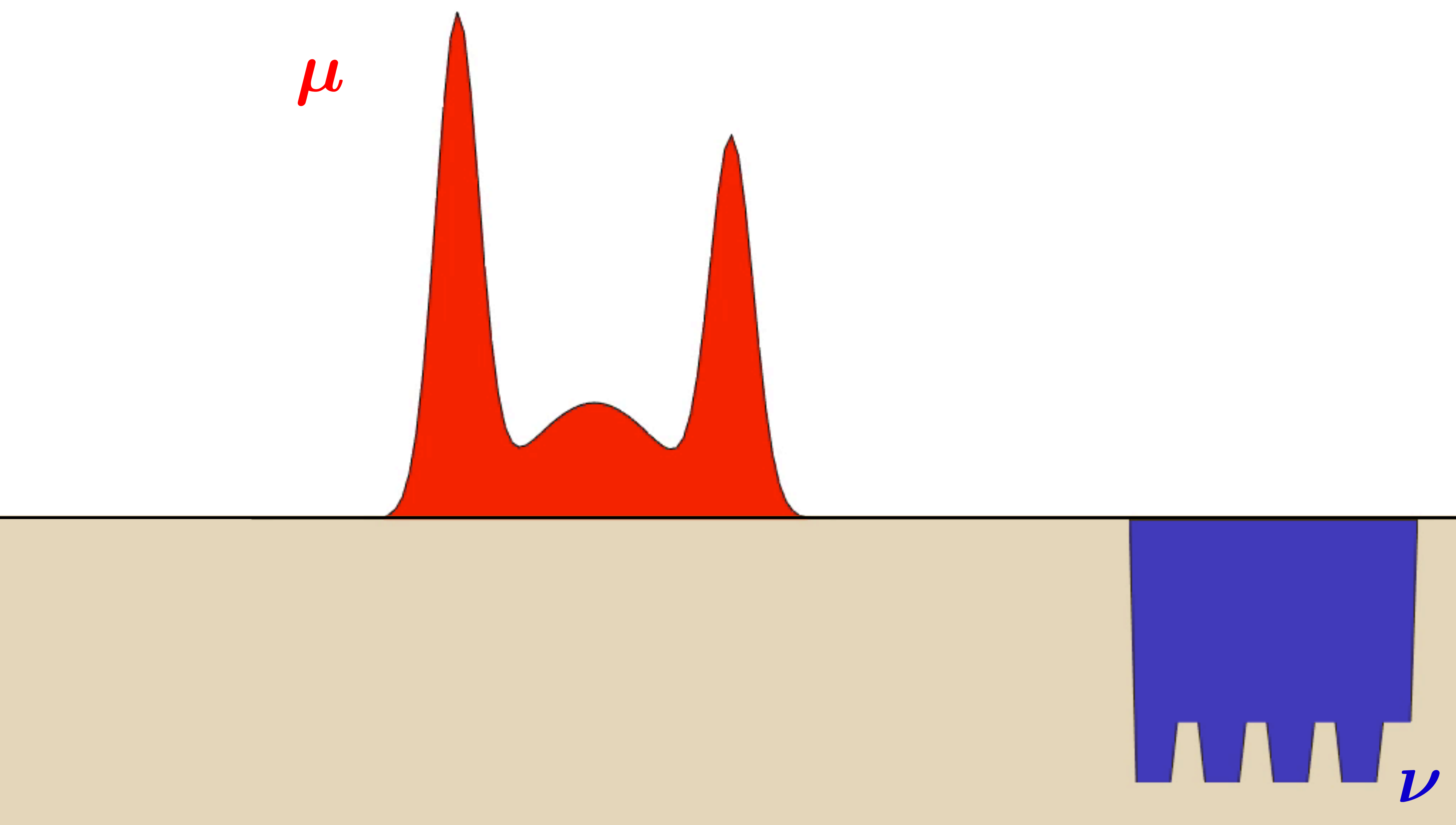
SUR LA

THÉORIE DES DÉBLAIS

When one has to bring earth from one place to another…

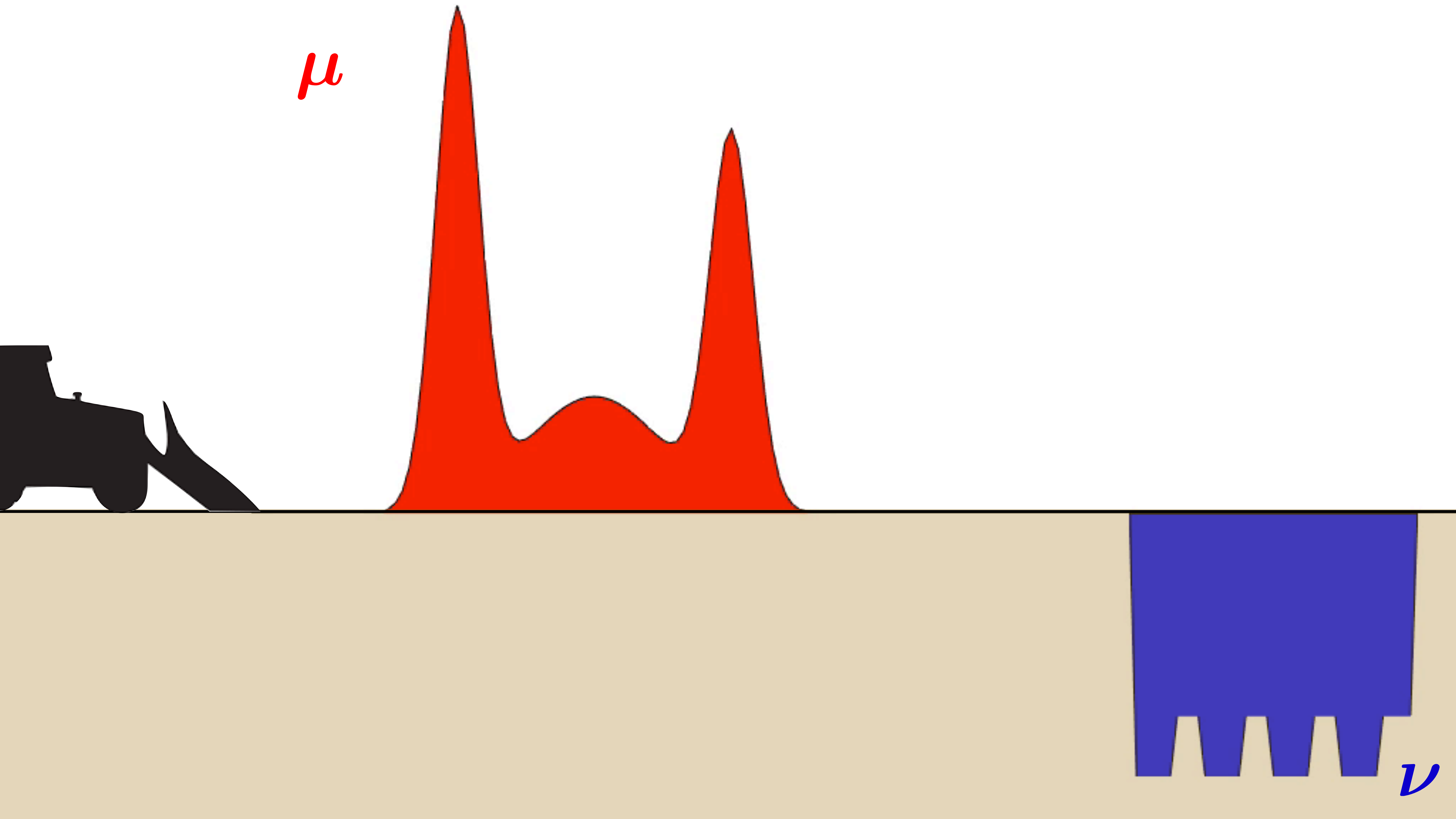
LORSQU'ON doit tranſporter des terres d'un lieu dans un autre, on a coutume de donner le nom de *Déblai* au volume des terres que l'on doit tranſporter, & le nom de *Remblai* à l'eſpace qu'elles doivent occuper après le tranſport.

Origins: Monge Problem



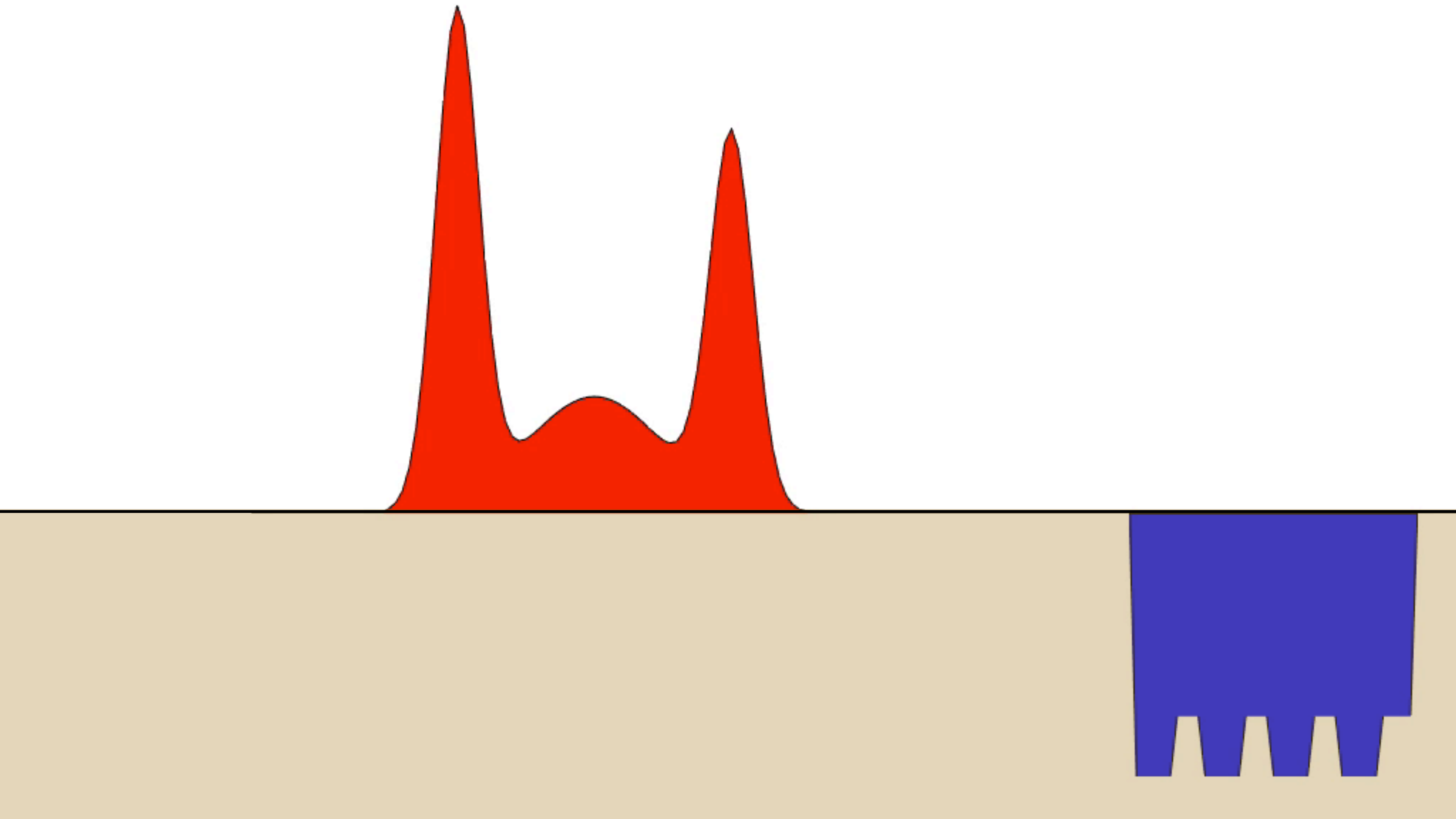
Origins: Monge Problem

In the 21st Century...



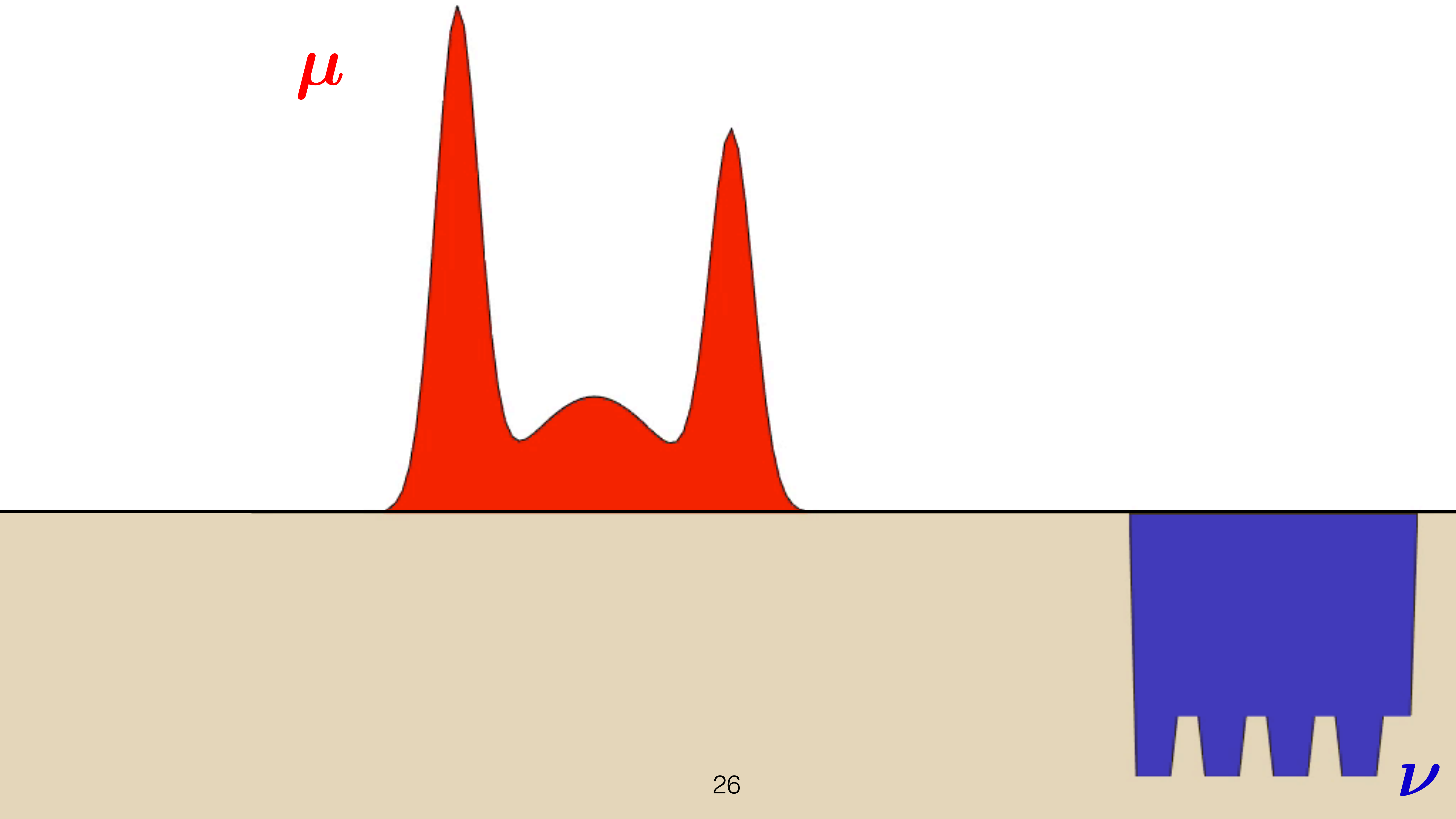
Origins: Monge Problem

In the 21st Century…



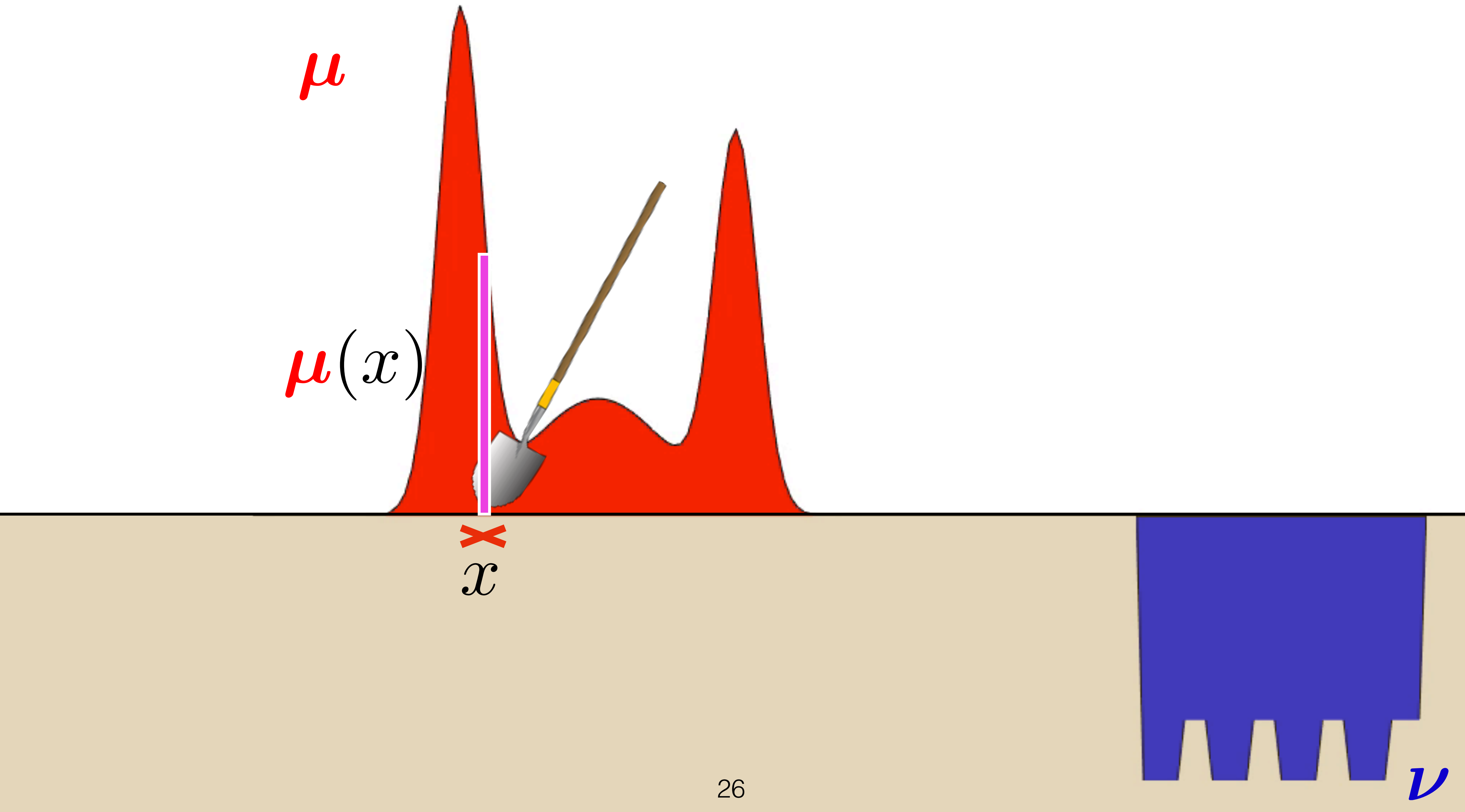
Origins: Monge's Problem

In 1781 however…



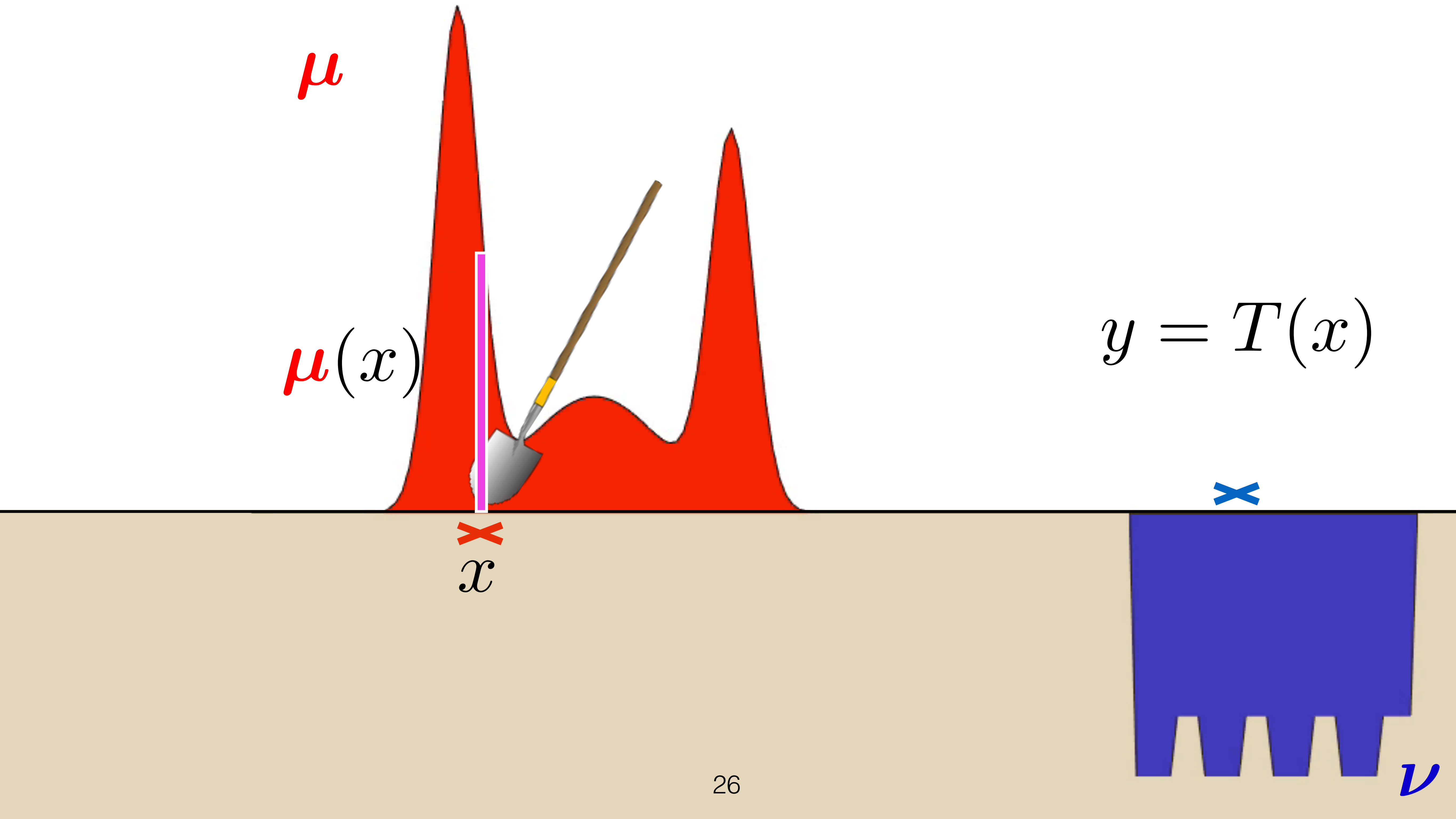
Origins: Monge's Problem

In 1781 however...



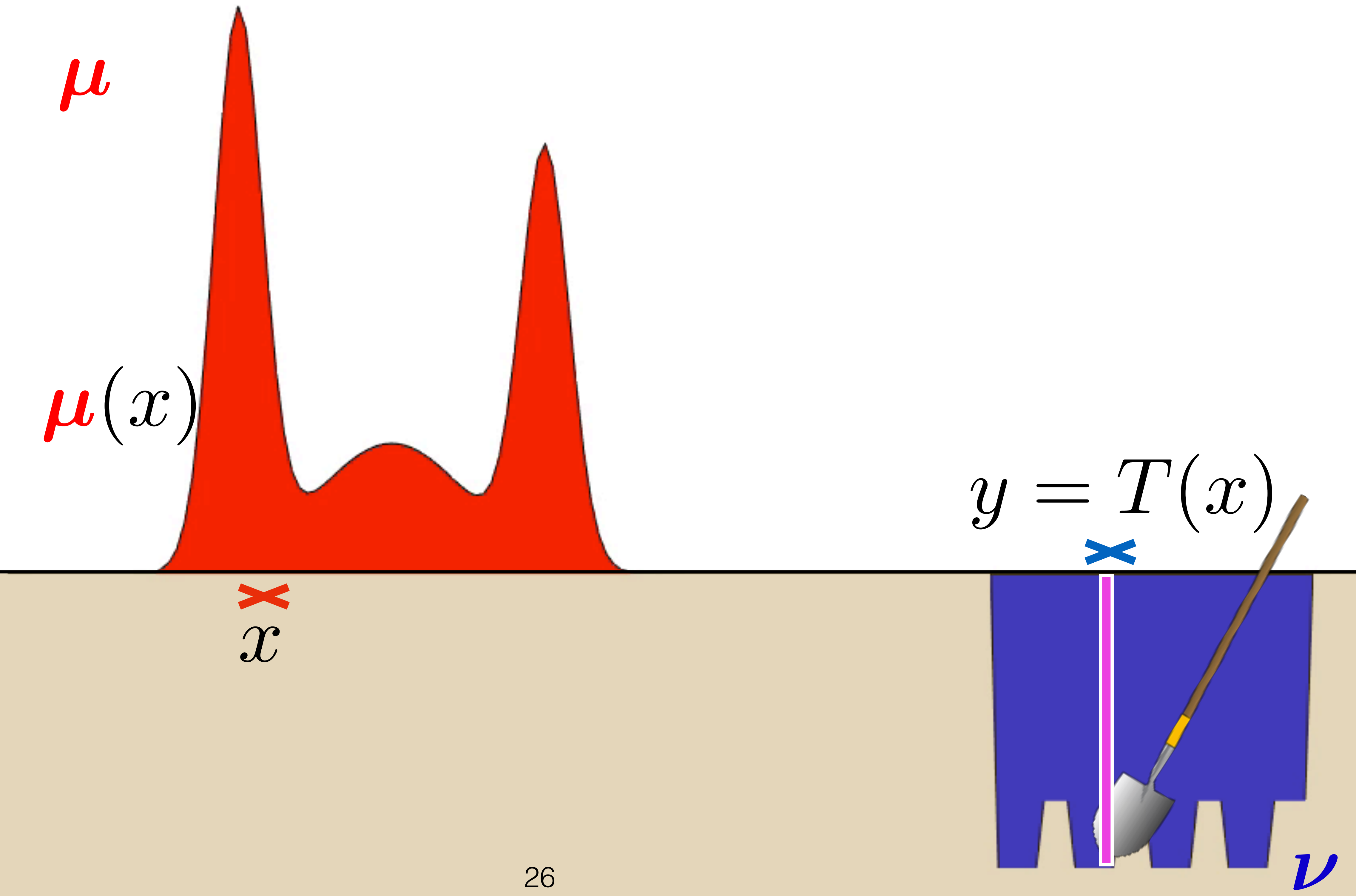
Origins: Monge's Problem

In 1781 however…



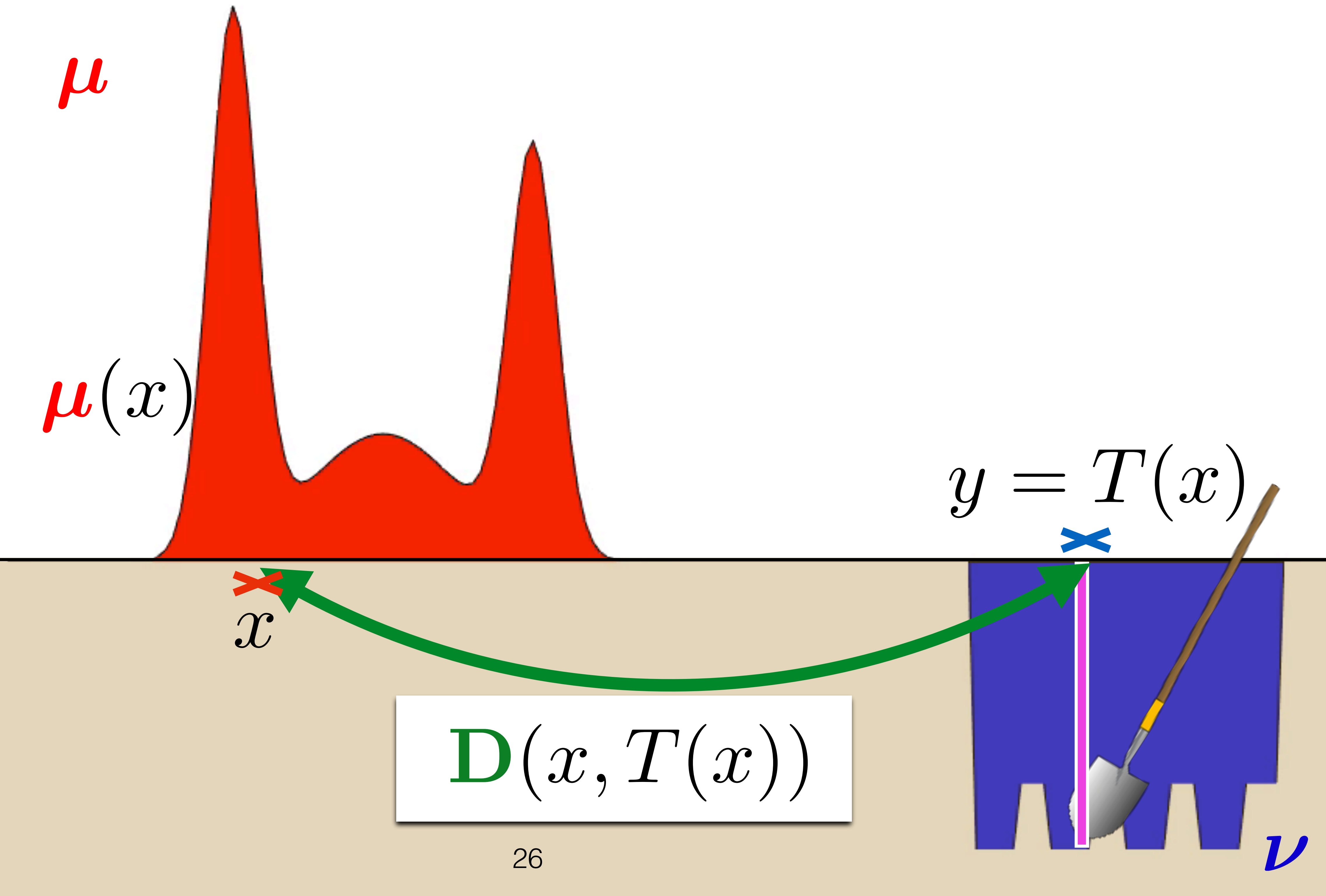
Origins: Monge's Problem

In 1781 however...



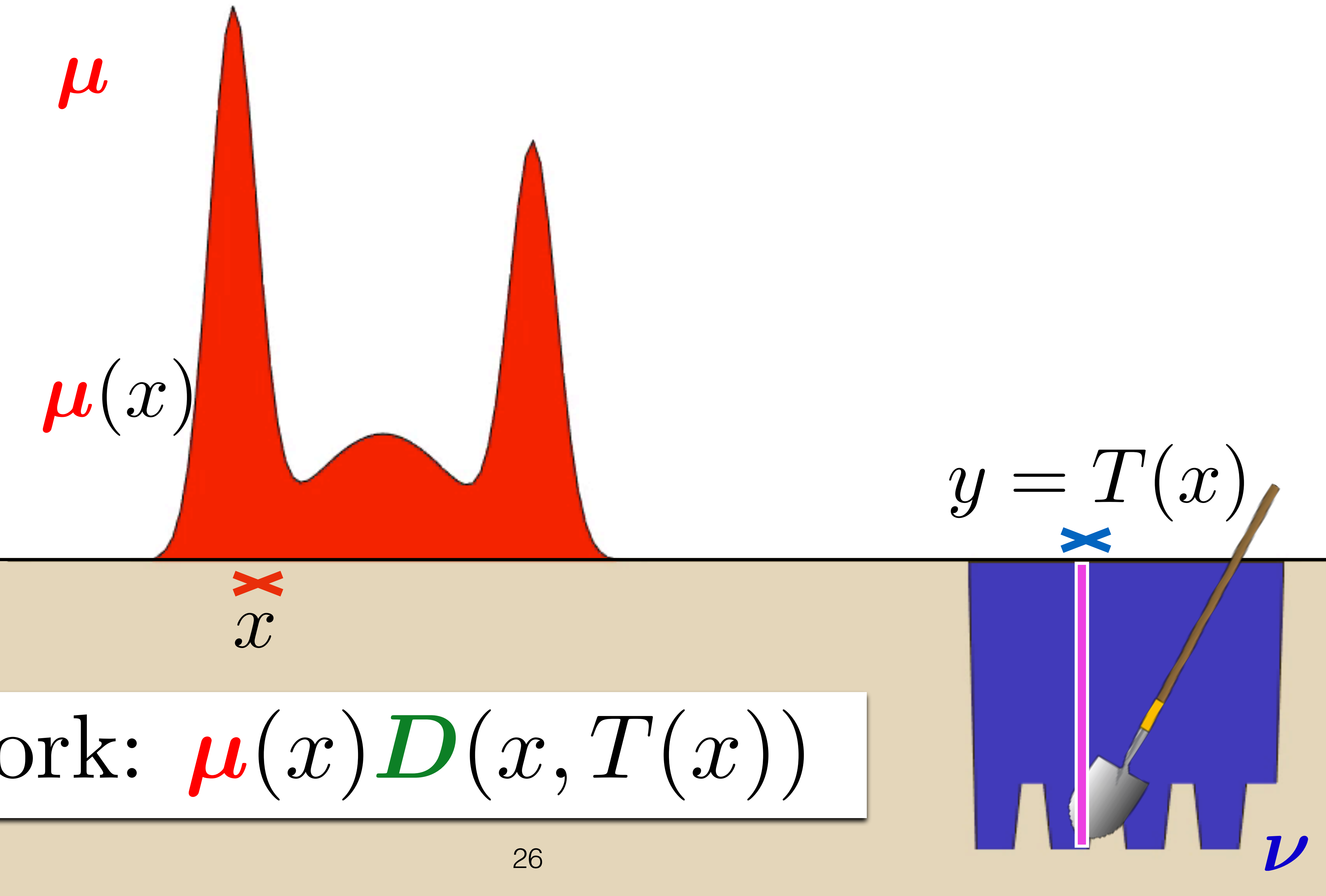
Origins: Monge's Problem

In 1781 however...



Origins: Monge's Problem

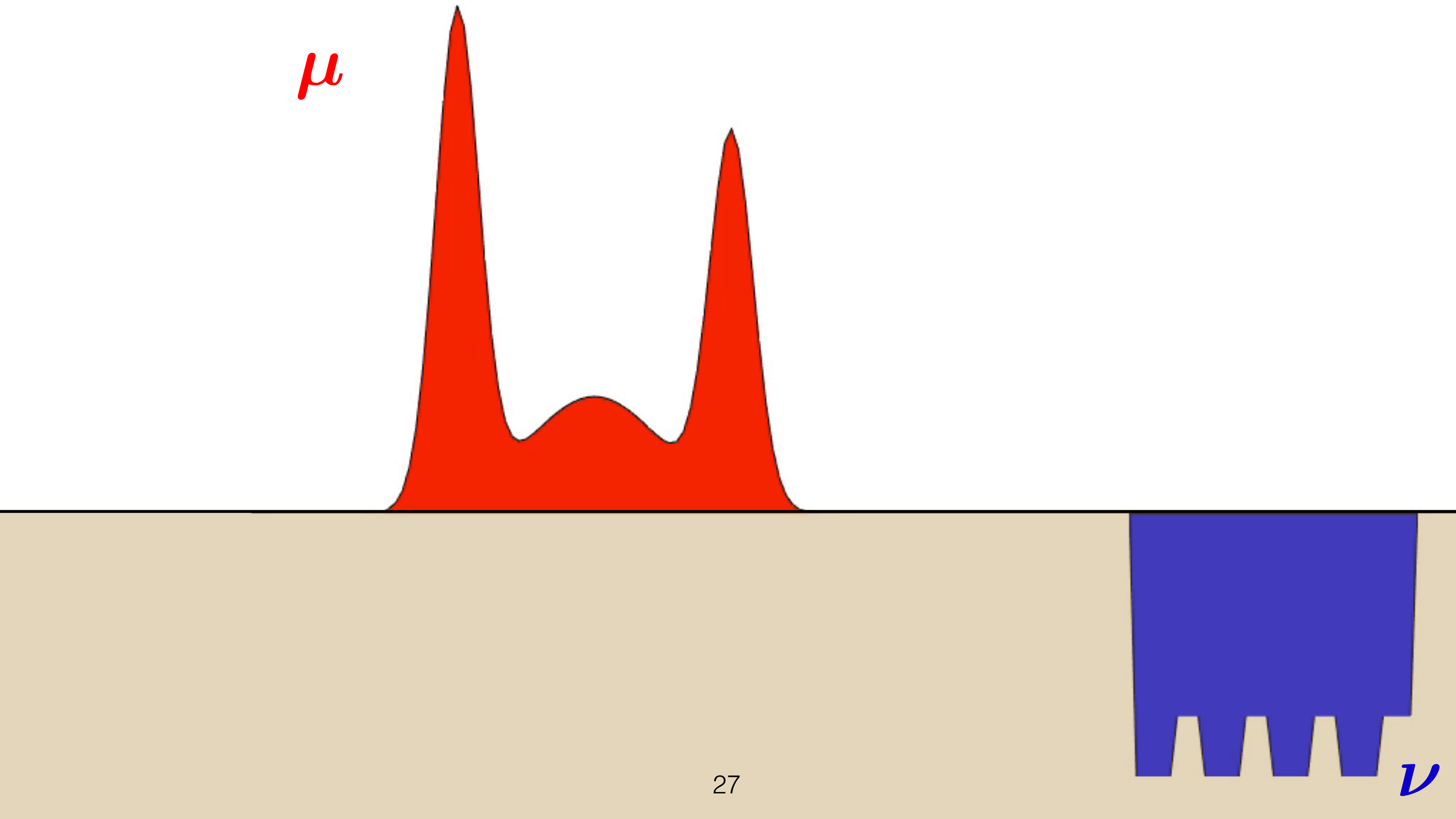
In 1781 however...



work: $\mu(x)D(x,T(x))$

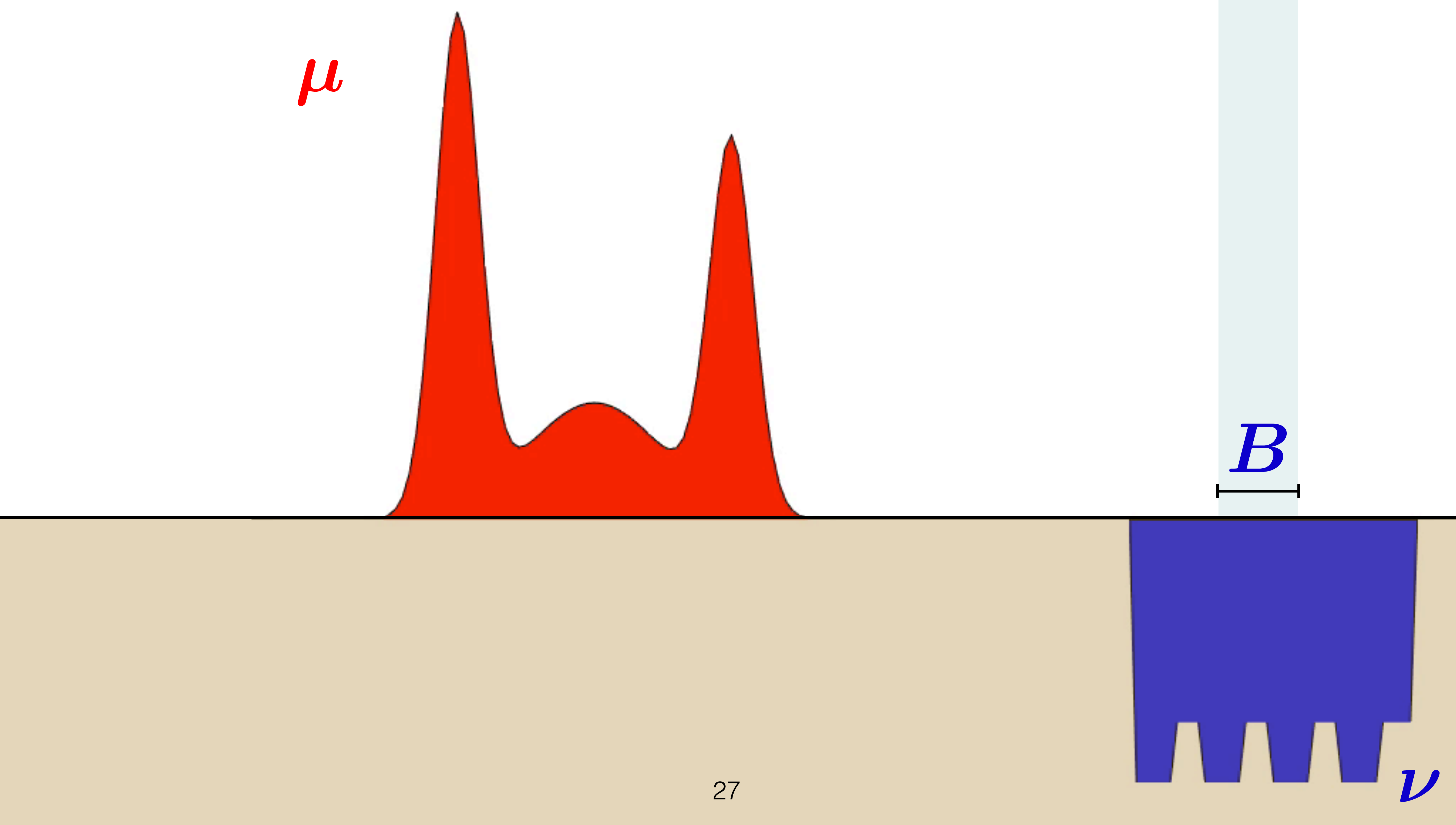
Origins: Monge's Problem

T must map red to blue.



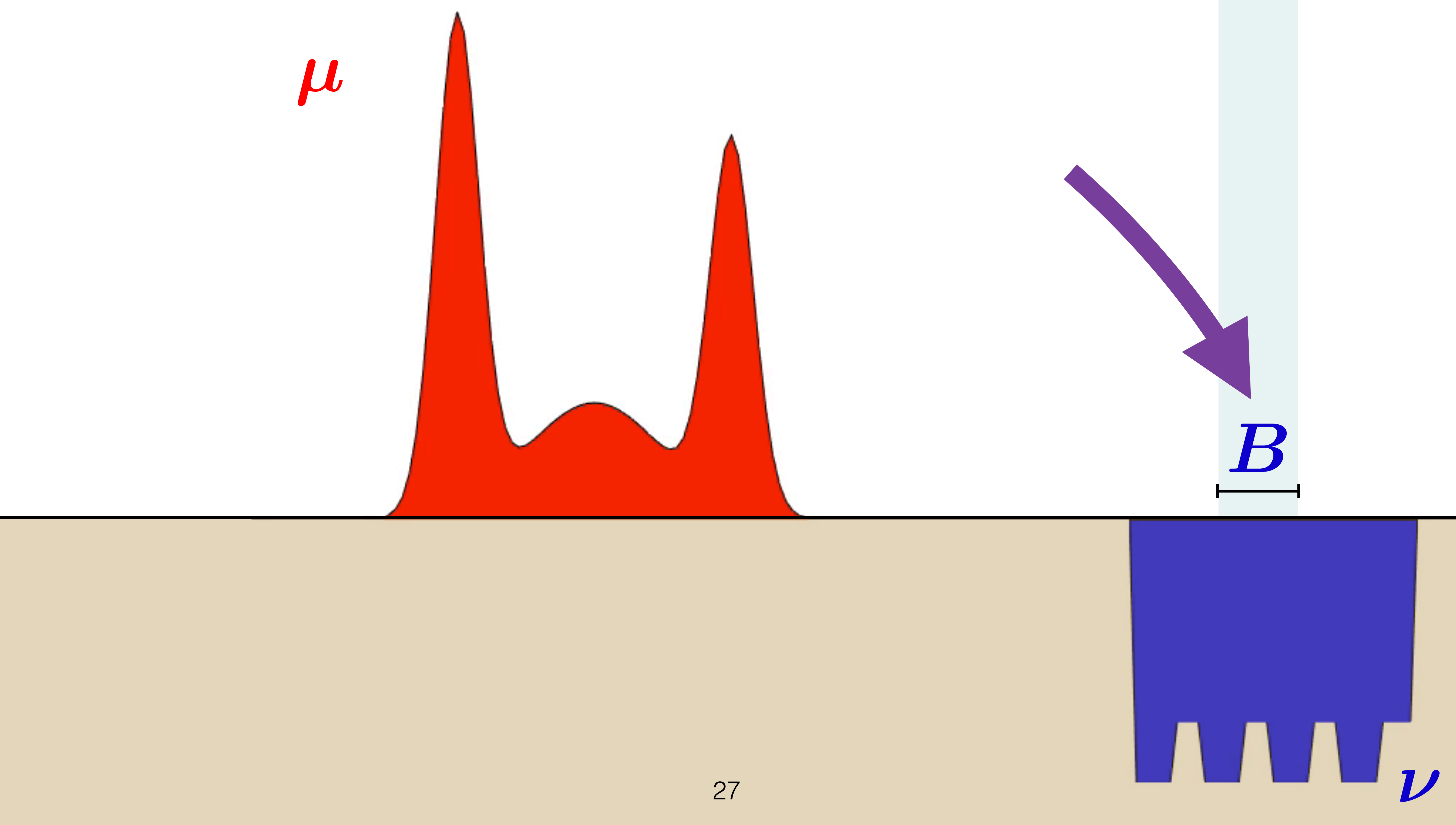
Origins: Monge's Problem

T must map red to blue.



Origins: Monge's Problem

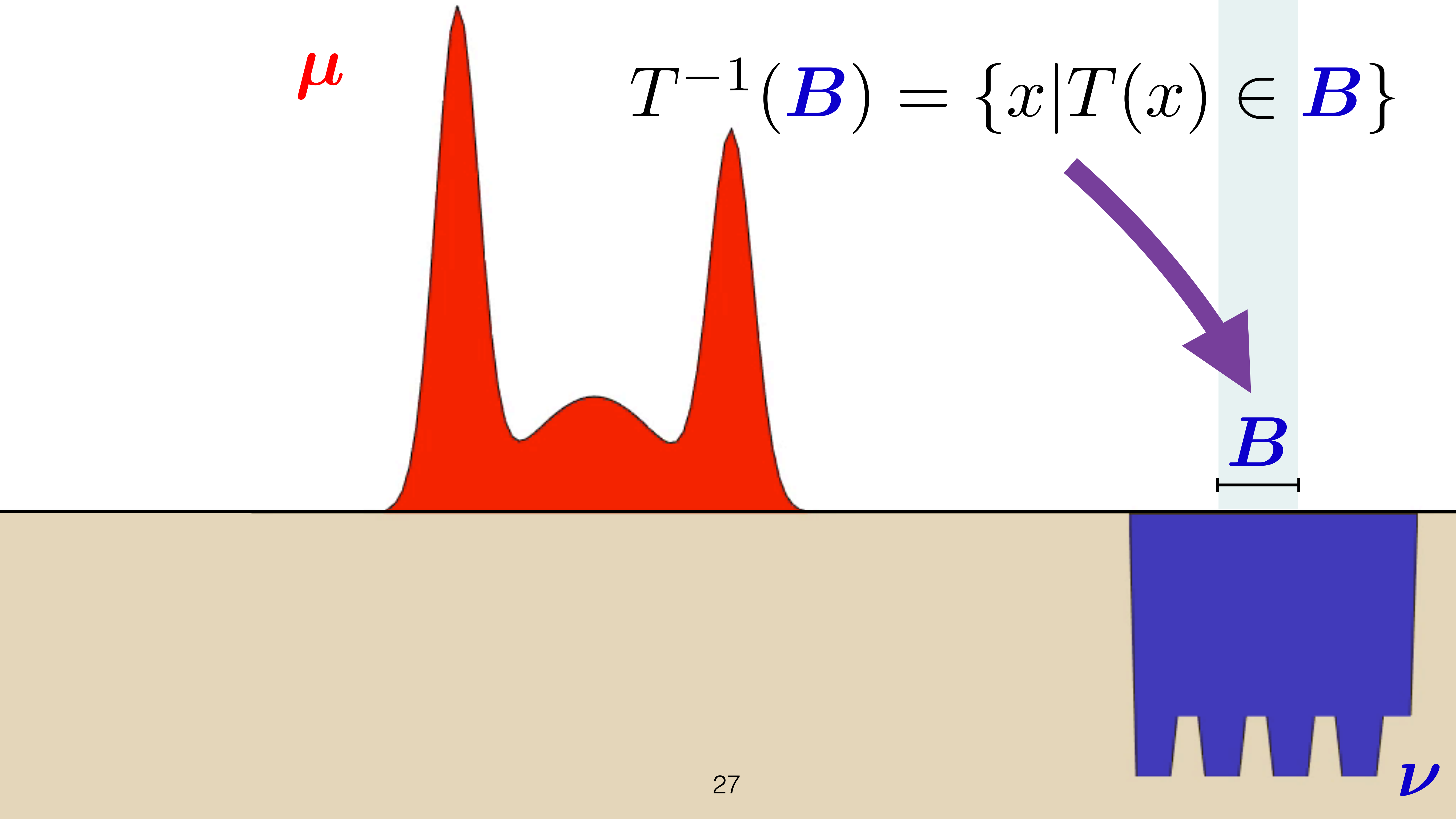
T must map red to blue.



Origins: Monge's Problem

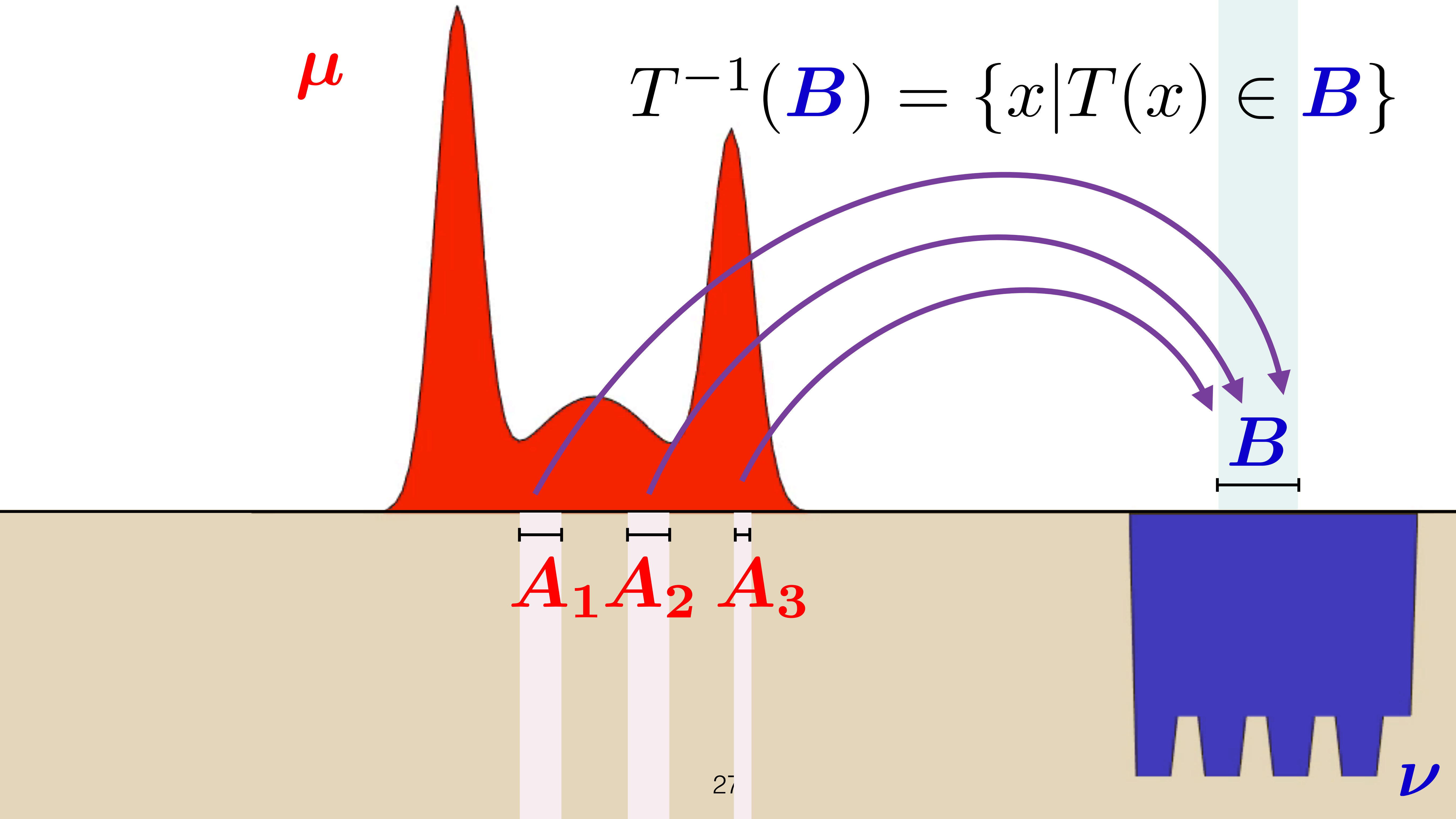
T must map red to blue.

$$T^{-1}(\boldsymbol{B}) = \{x | T(x) \in \boldsymbol{B}\}$$



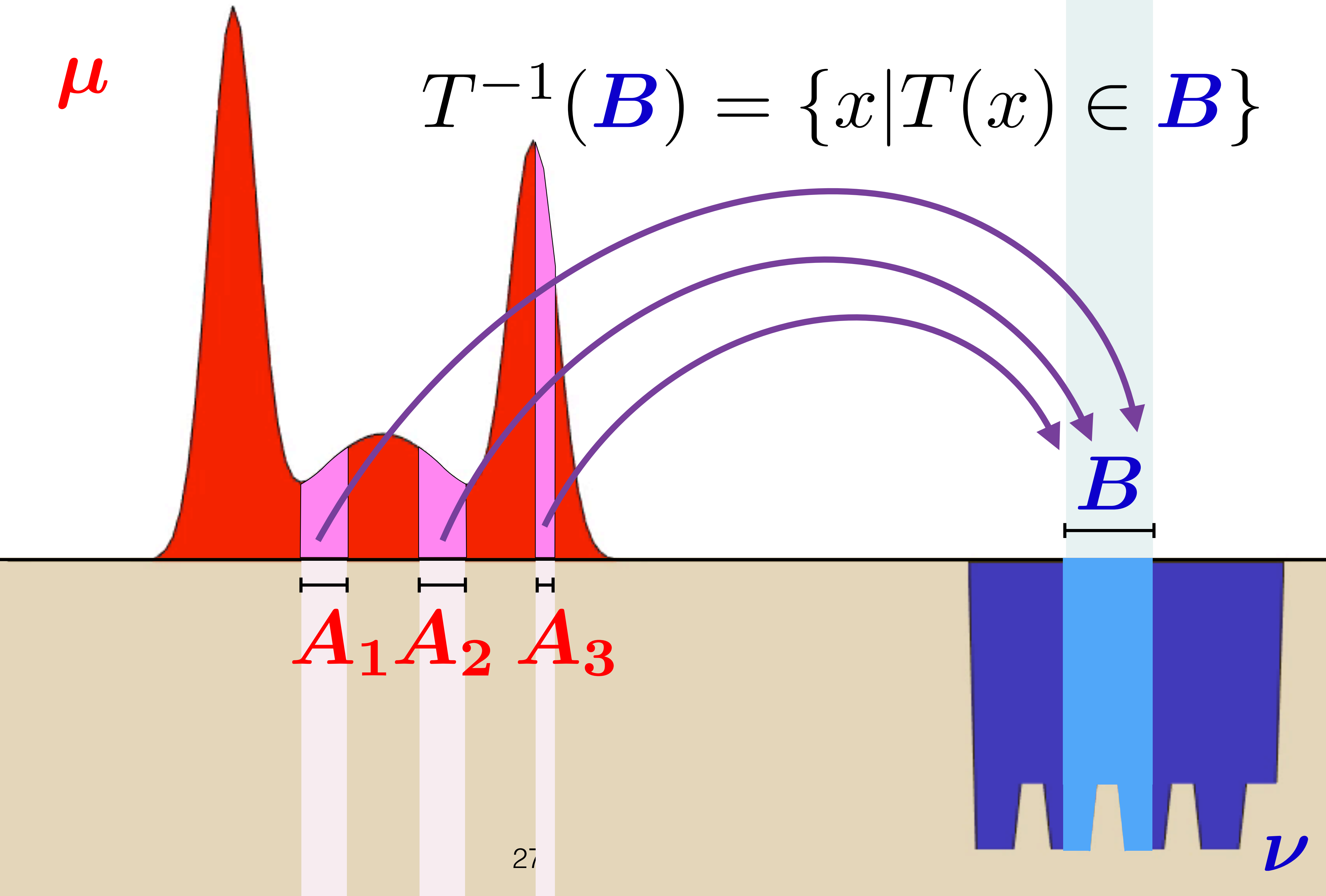
Origins: Monge's Problem

T must map red to blue.



Origins: Monge's Problem

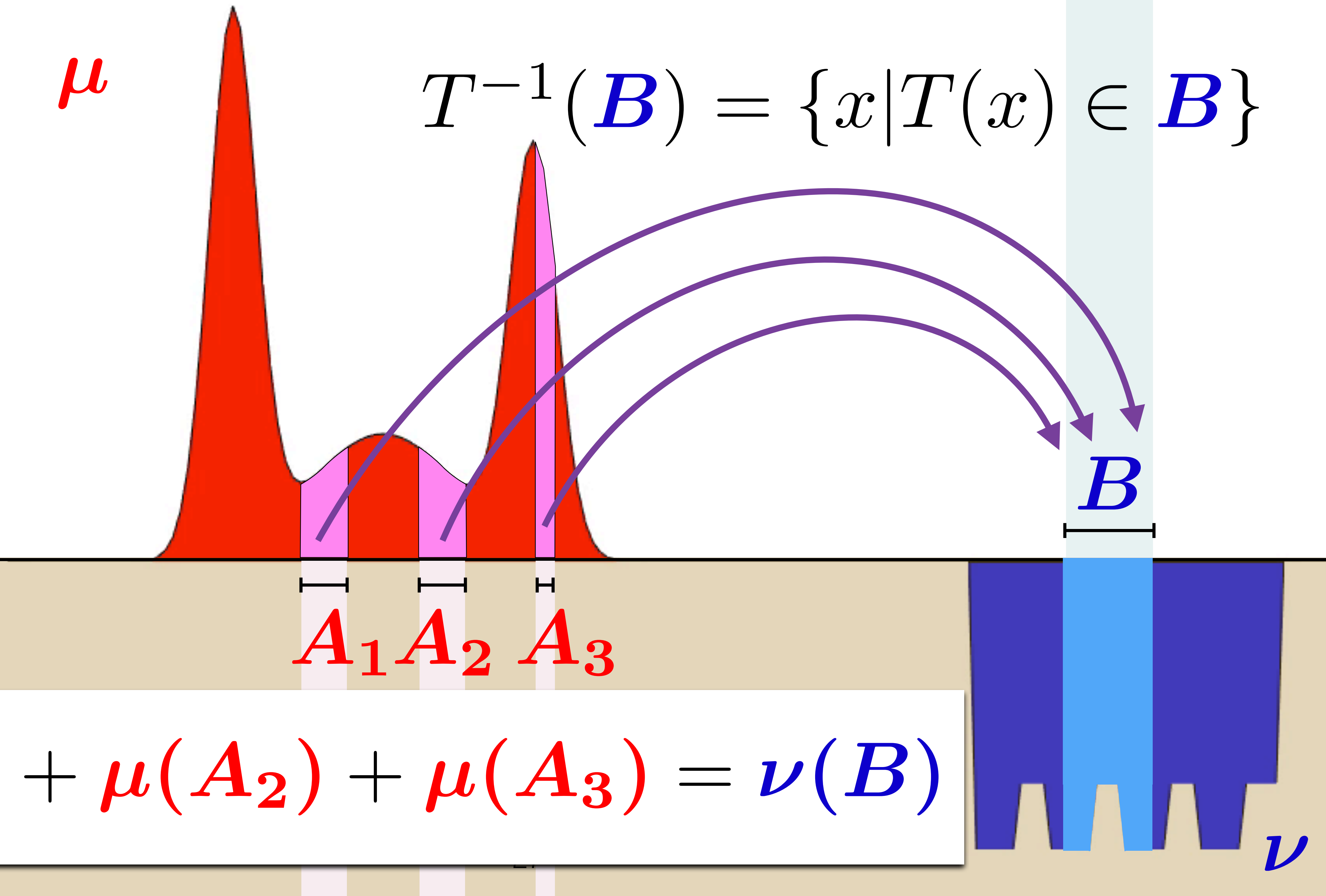
T must map red to blue.



Origins: Monge's Problem

T must map red to blue.

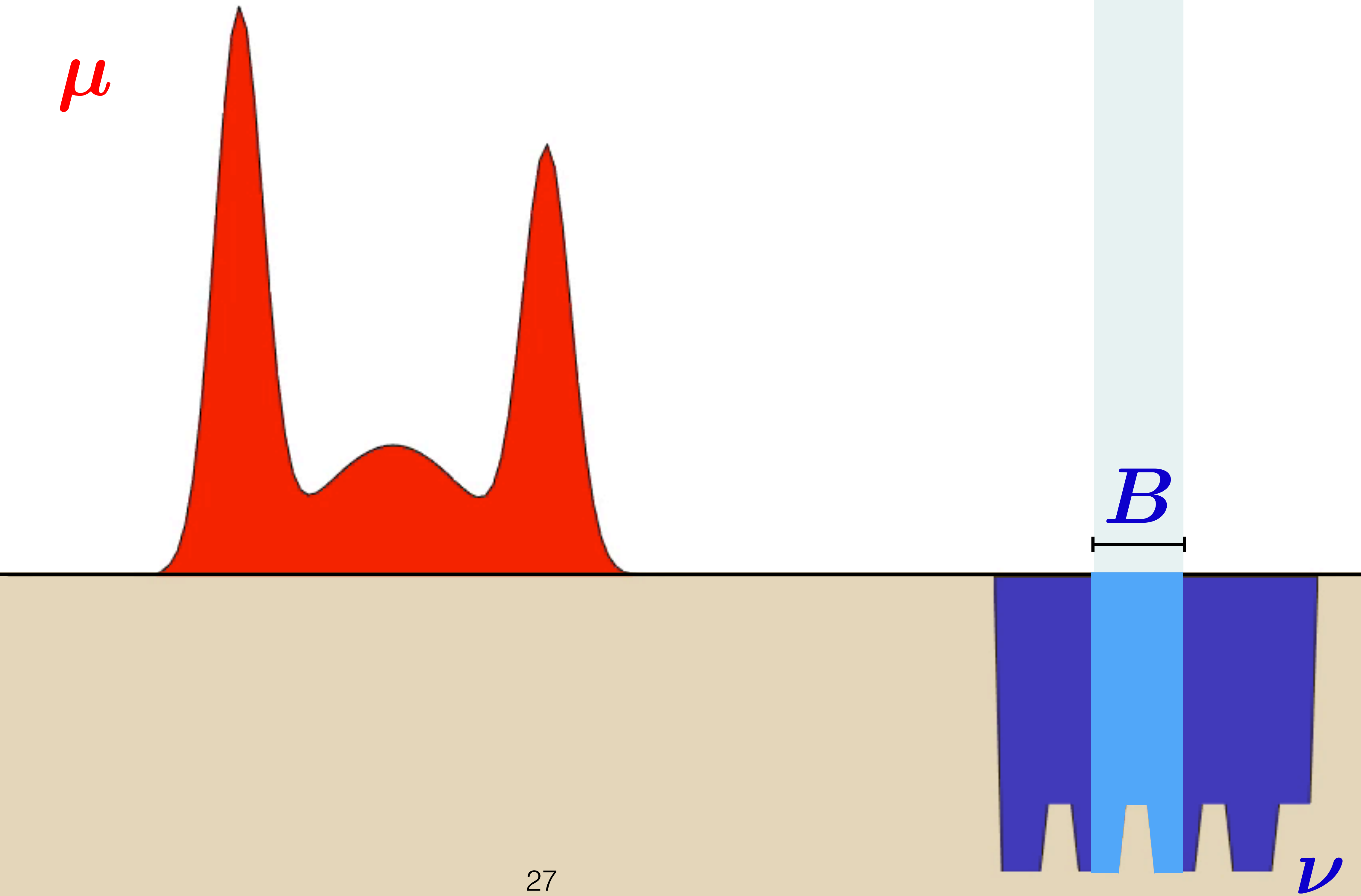
$$T^{-1}(\boldsymbol{B}) = \{x | T(x) \in \boldsymbol{B}\}$$



$$\boldsymbol{\mu}(\boldsymbol{A_1}) + \boldsymbol{\mu}(\boldsymbol{A_2}) + \boldsymbol{\mu}(\boldsymbol{A_3}) = \boldsymbol{\nu}(\boldsymbol{B})$$

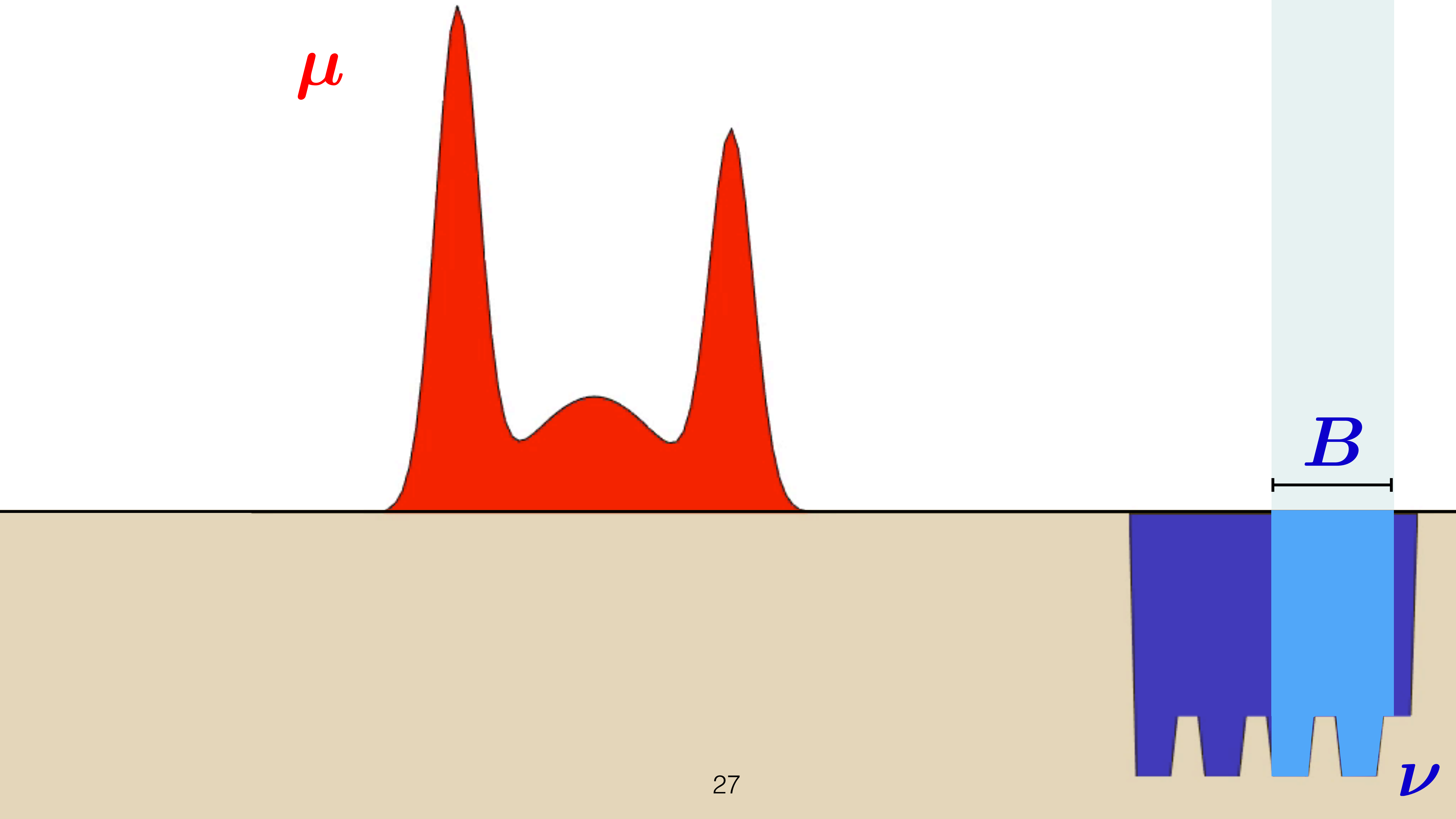
Origins: Monge's Problem

T must map red to blue.



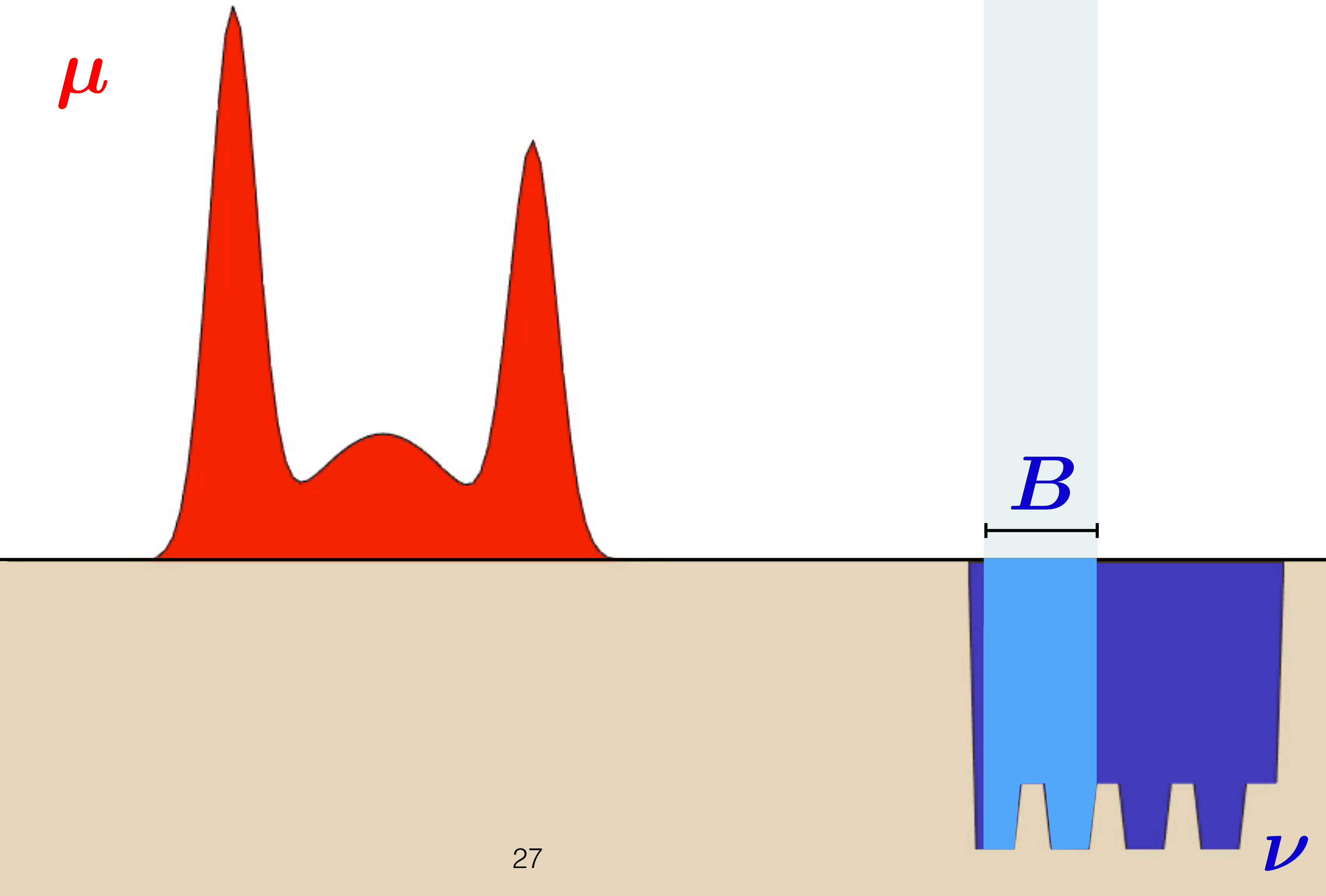
Origins: Monge's Problem

T must map red to blue.



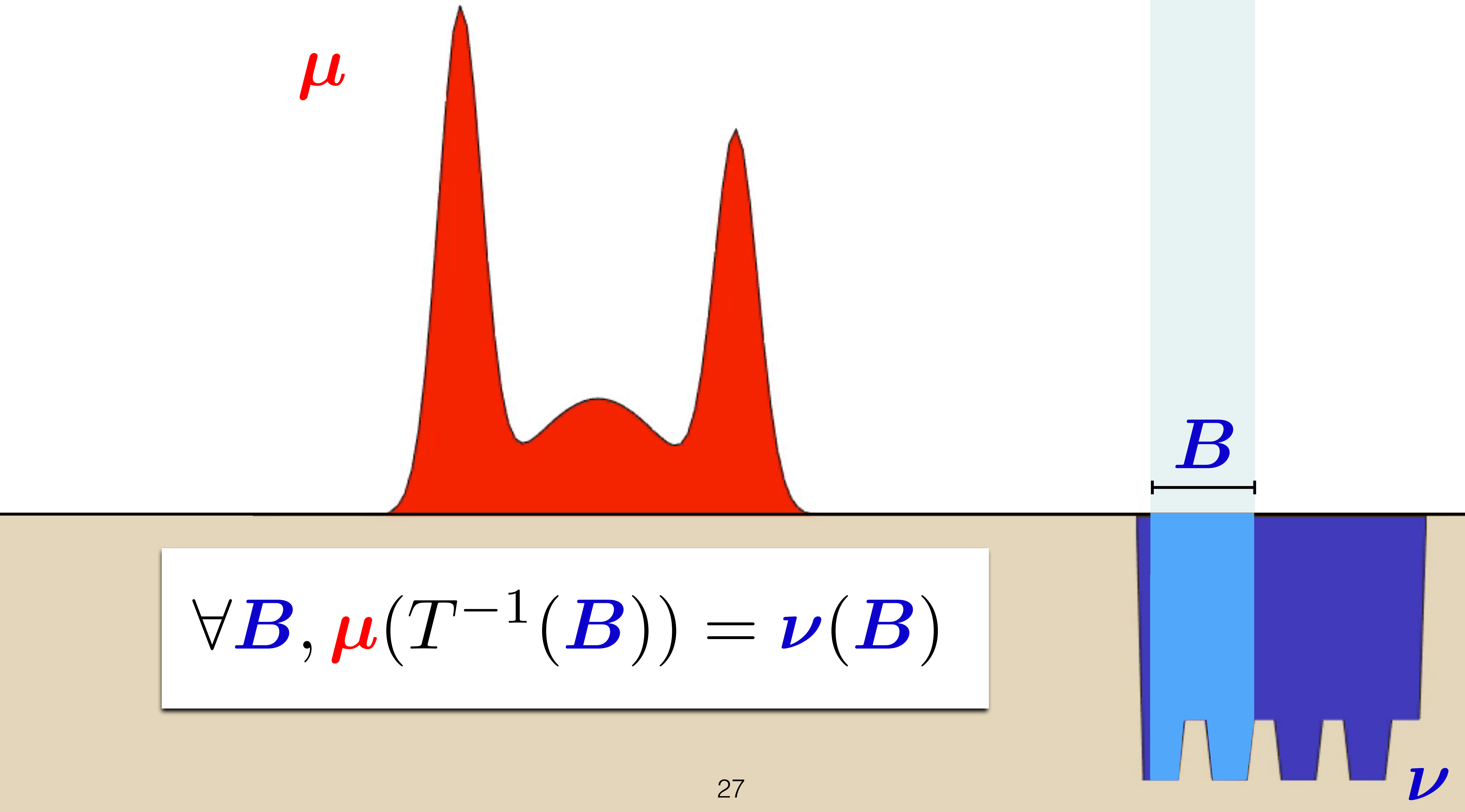
Origins: Monge's Proble

T must map red to blue.



Origins: Monge's Proble

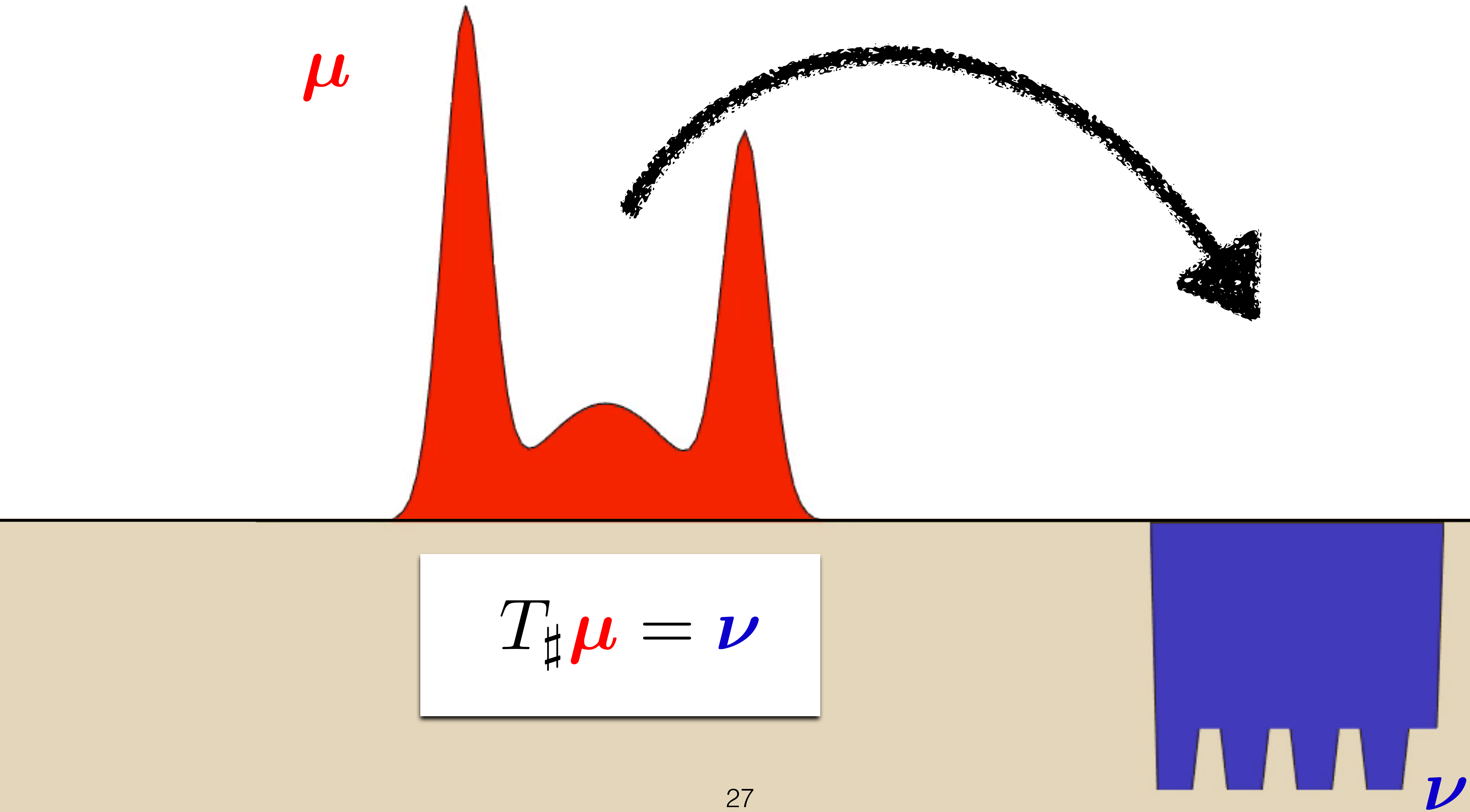
T must map red to blue.



$$\forall B, \mu(T^{-1}(B)) = \nu(B)$$

Origins: Monge's Problem

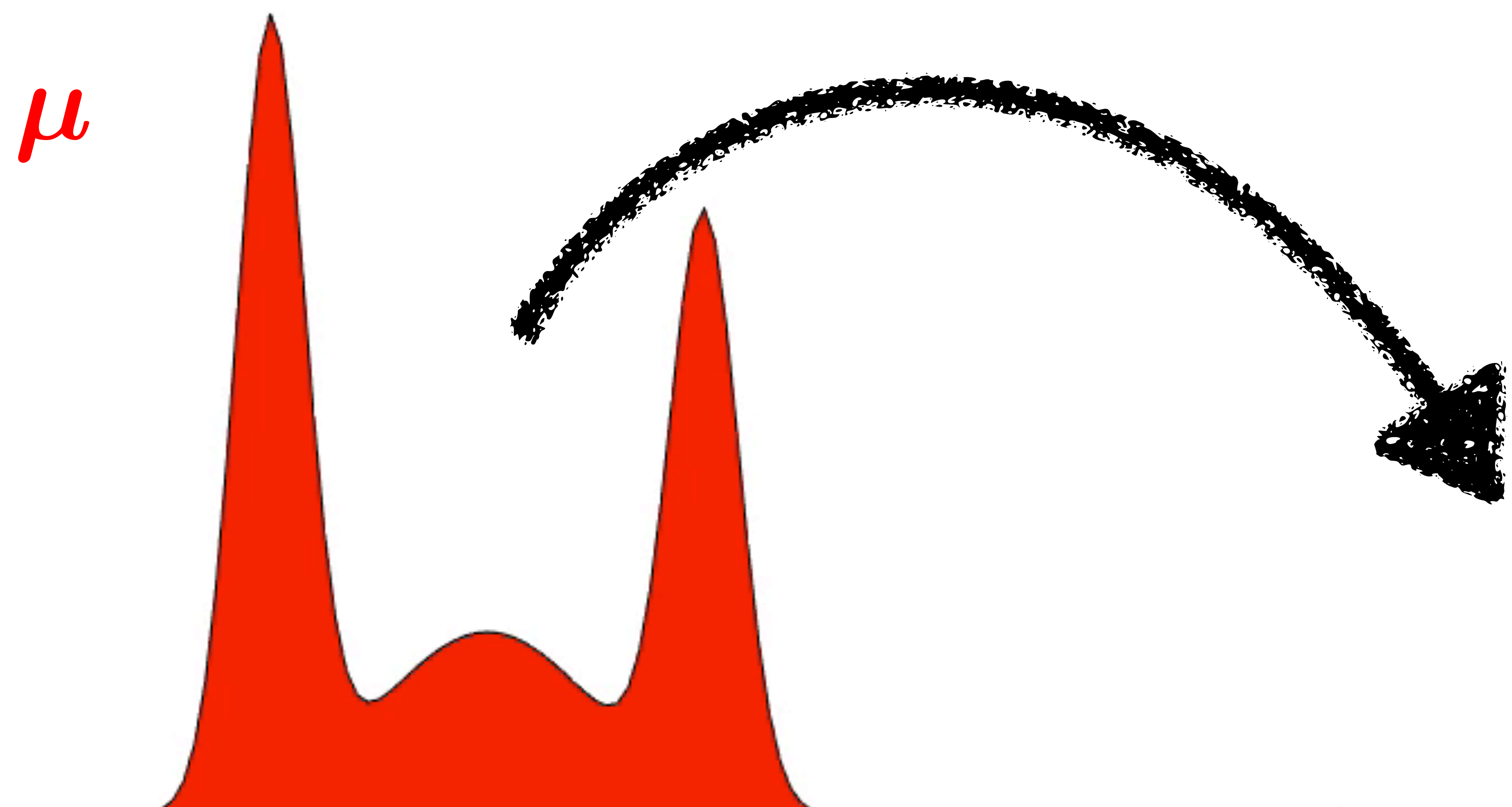
T must **push-forward** *the red measure towards the blue*



$$T_\sharp \mu = \nu$$

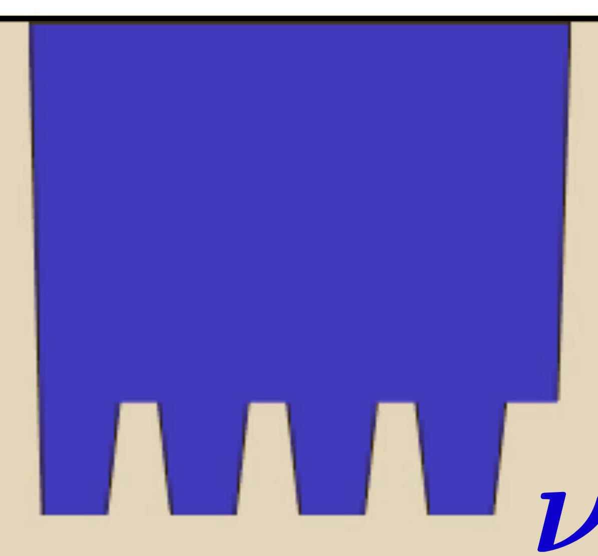
Origins: Monge's Problem

T must **push-forward** *the red measure towards the blue*



What T s.t. $T_\sharp \boldsymbol{\mu} = \boldsymbol{\nu}$

minimizes $\int \boldsymbol{D}(x, T(x)) \boldsymbol{\mu}(dx)$?



Kantorovich Problem

Kantorovich

Tolstoi

1930

Hitchcock

1939

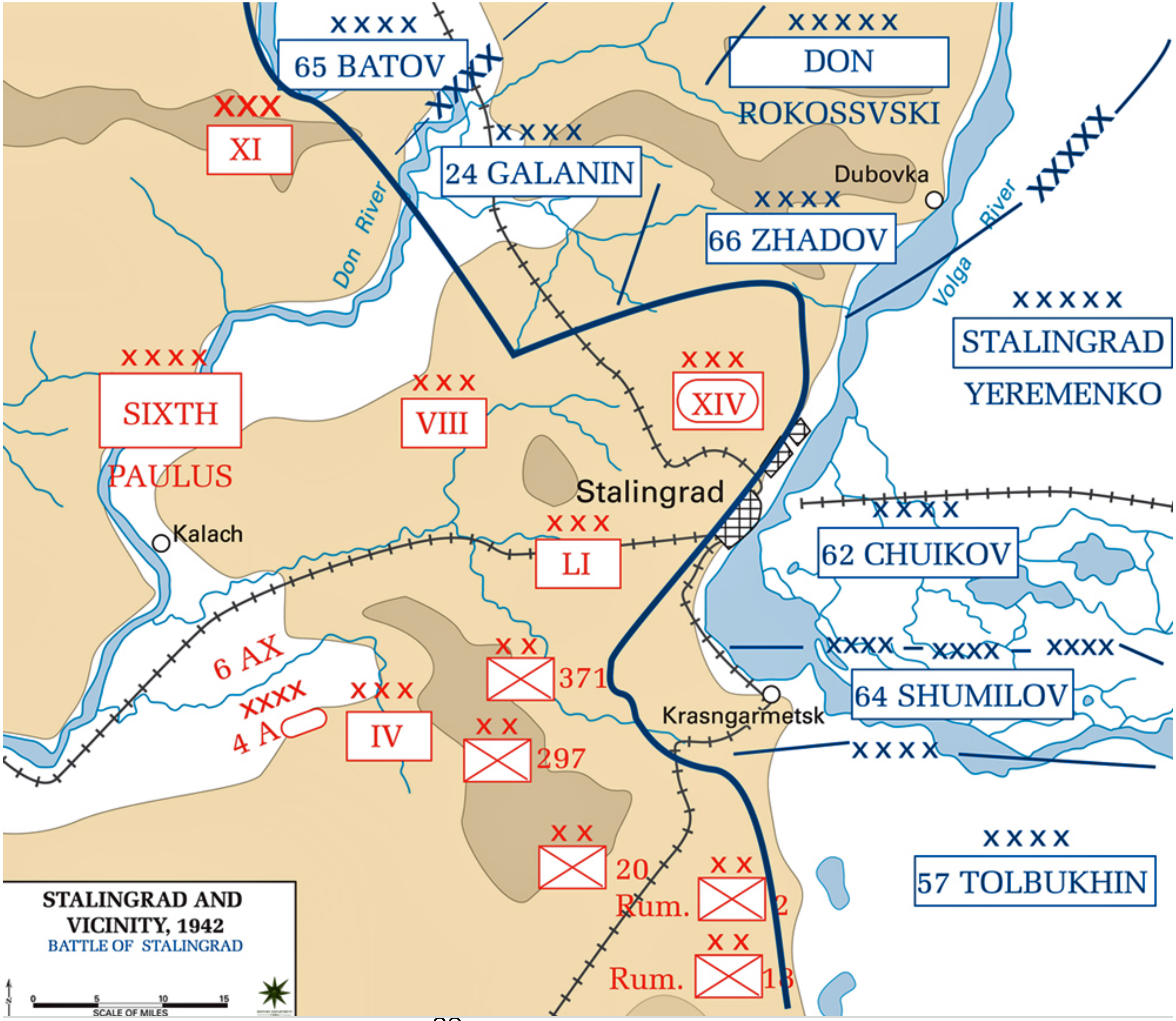
THE DISTRIBUTION OF A PRODUCT FROM SEVERAL SOURCES TO NUMEROUS LOCALITIES

BY FRANK L. HITCHCOCK

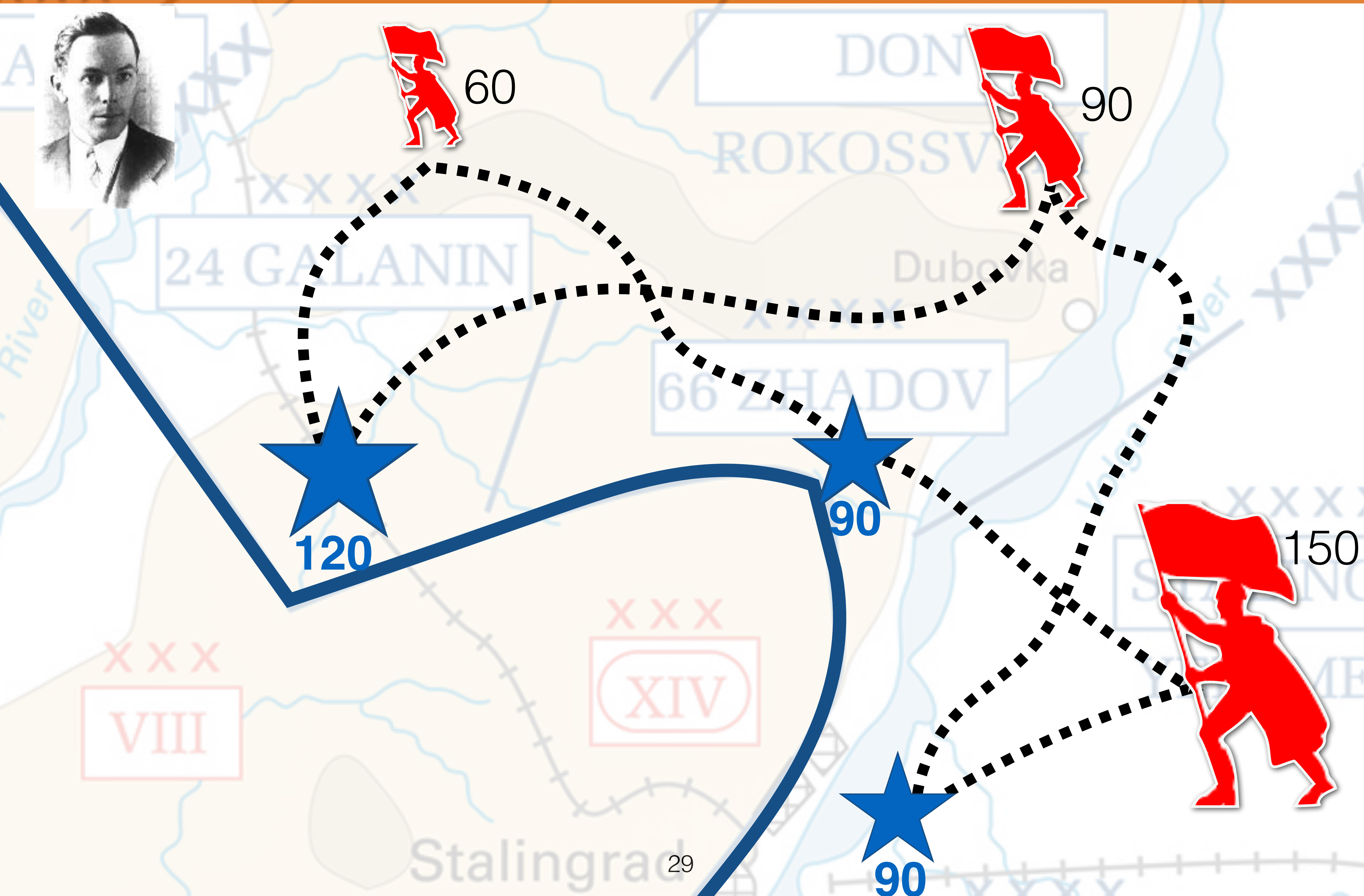
1. Statement of the problem. When several factories supply a product to a number of cities we desire the least costly manner of distribution. Due to freight rates and other matters the cost of a ton of product to a particular city will vary according to which factory supplies it, and will also vary from city to city.

1941

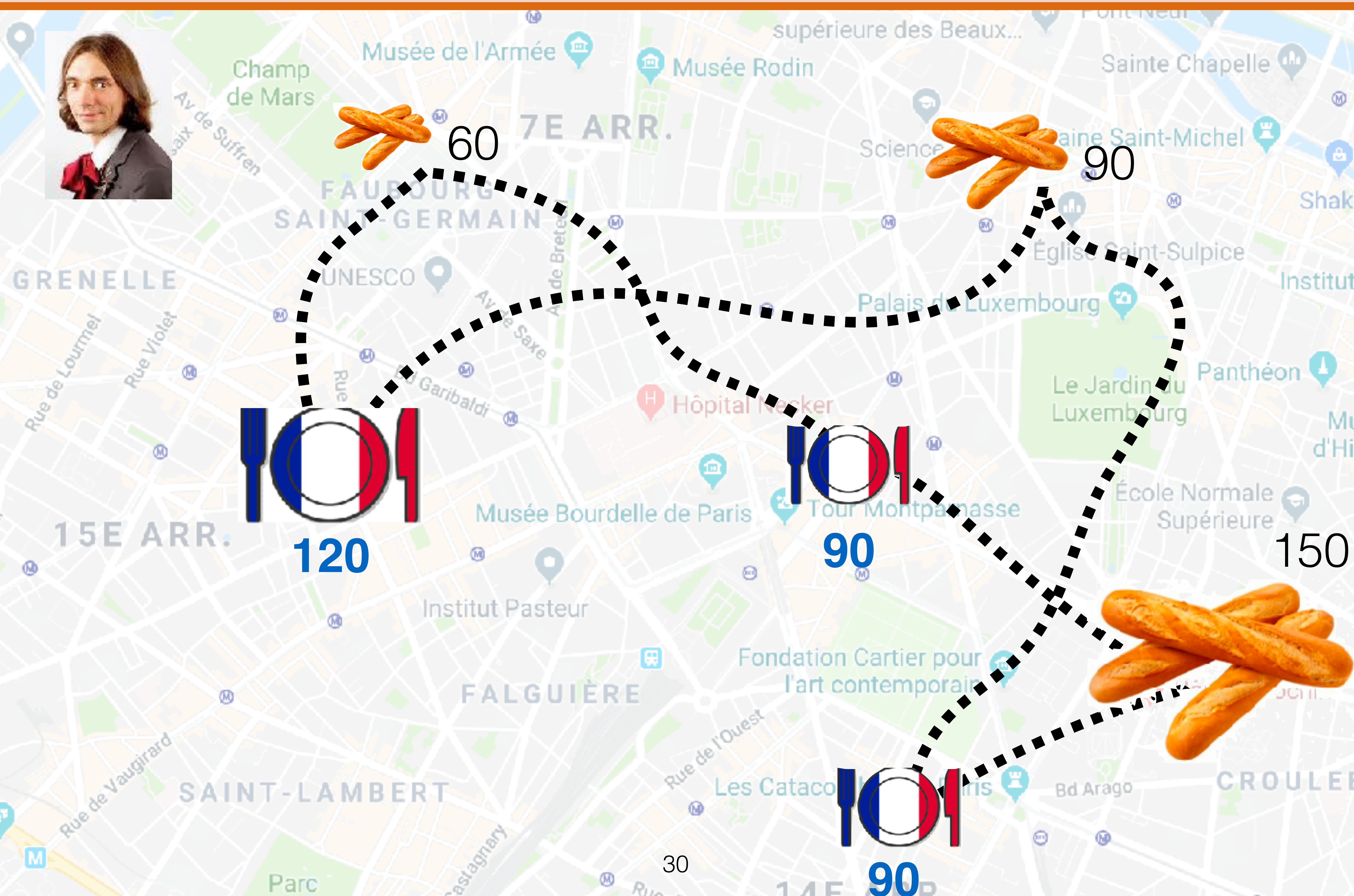
Kantorovich Problem



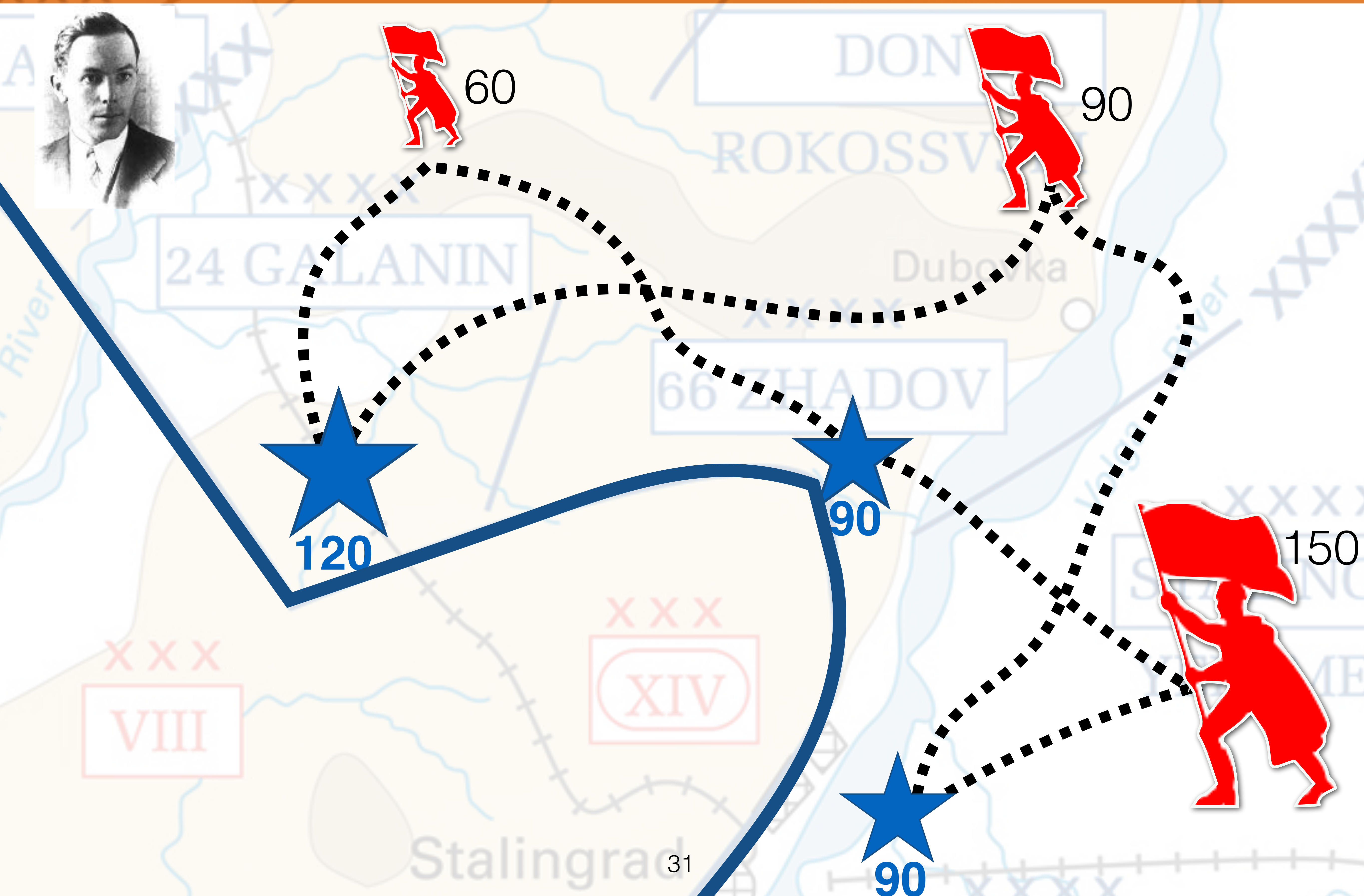
Kantorovich Problem



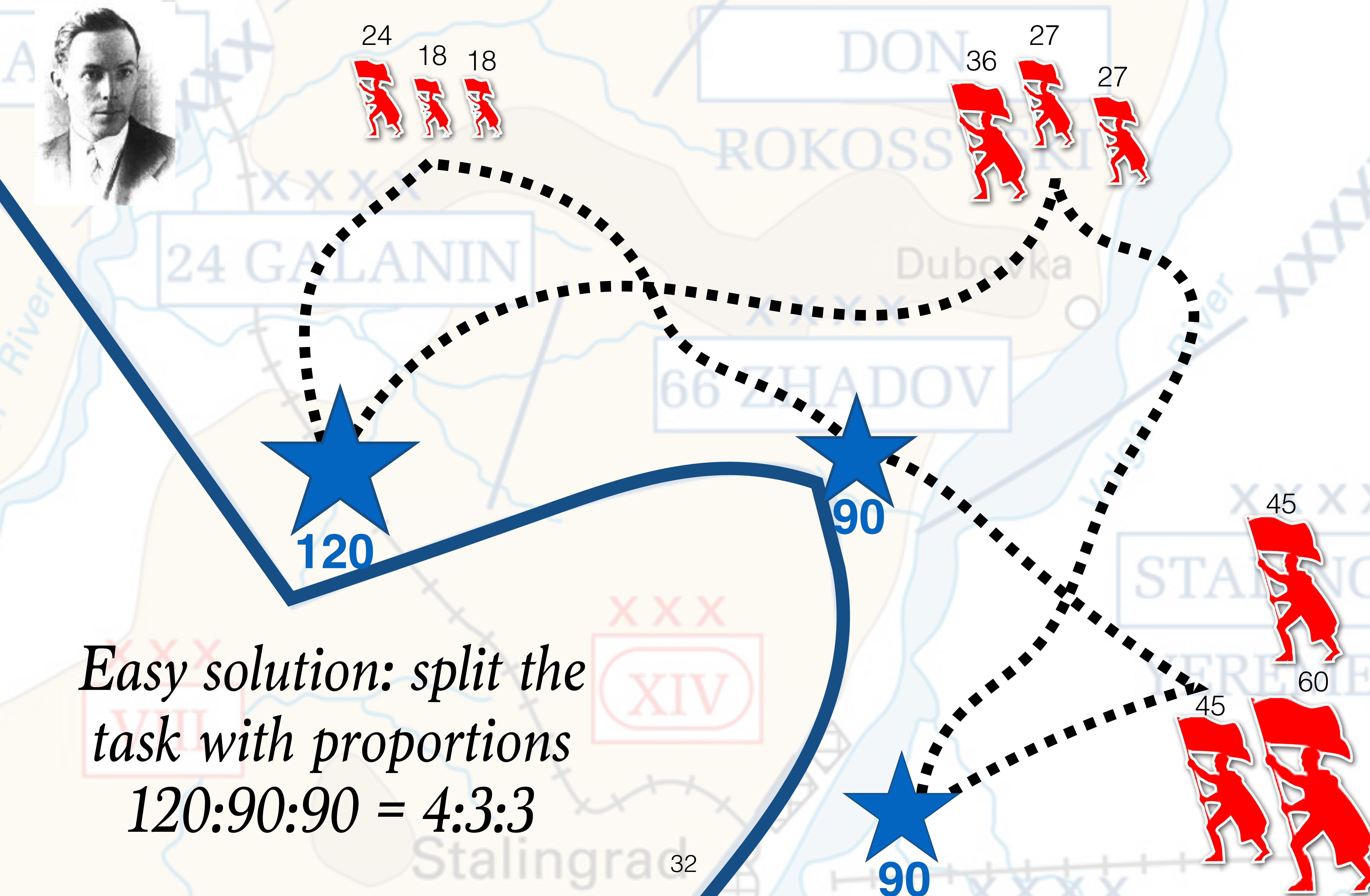
Kantorovich Problem *à la française*



Kantorovich Problem

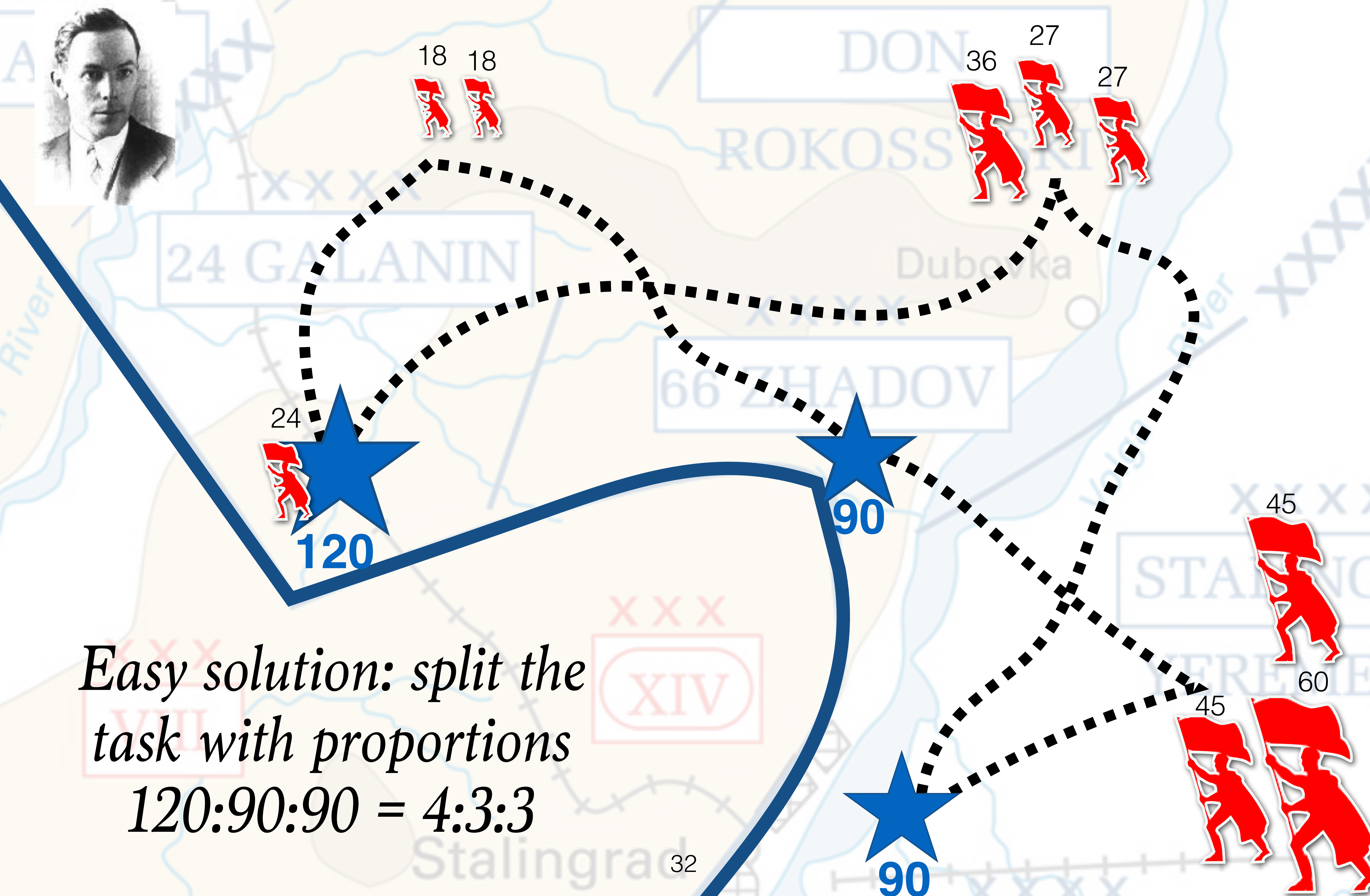


Kantorovich Problem



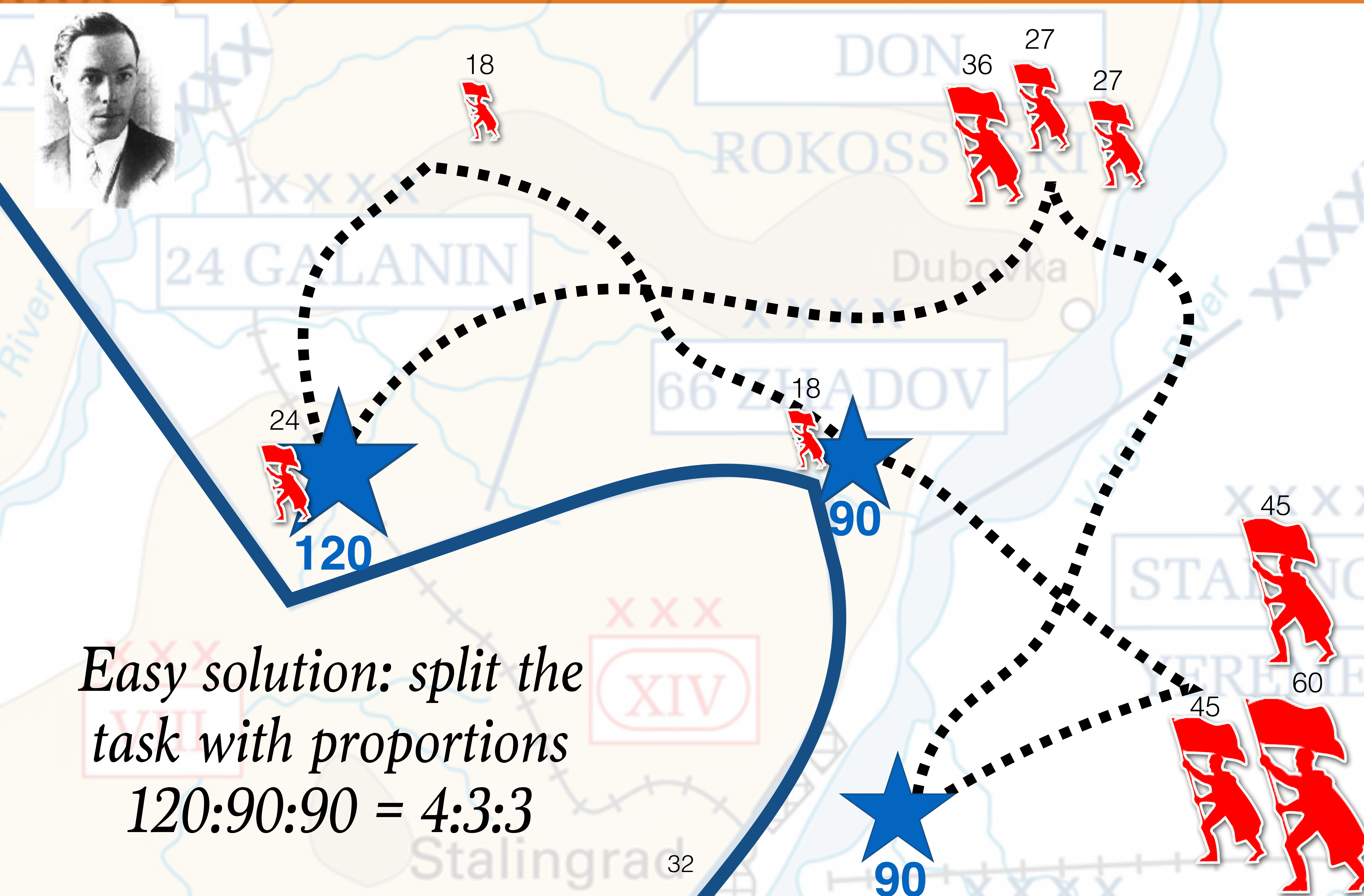
Easy solution: split the task with proportions 120:90:90 = 4:3:3

Kantorovich Problem



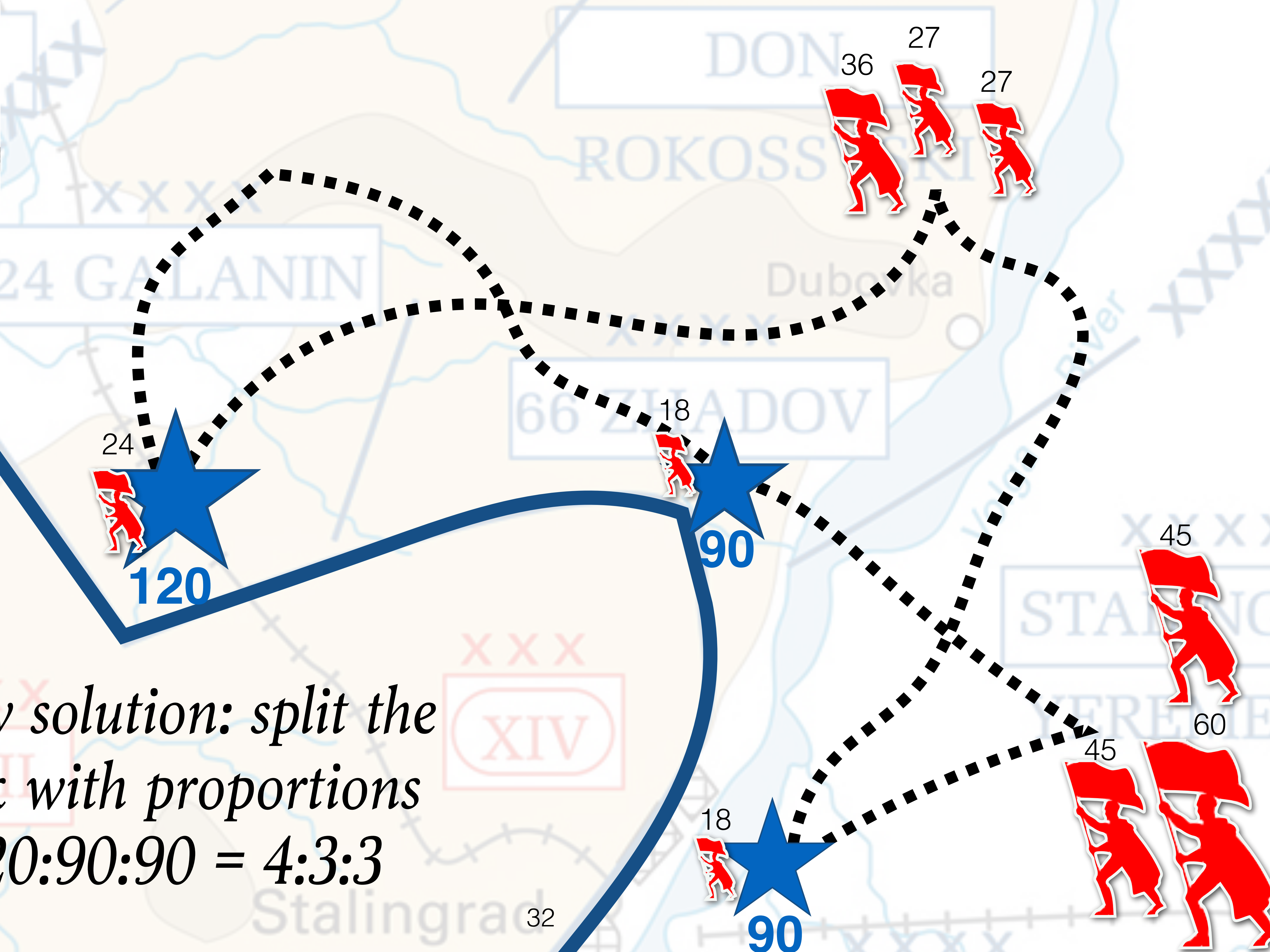
Easy solution: split the task with proportions 120:90:90 = 4:3:3

Kantorovich Problem



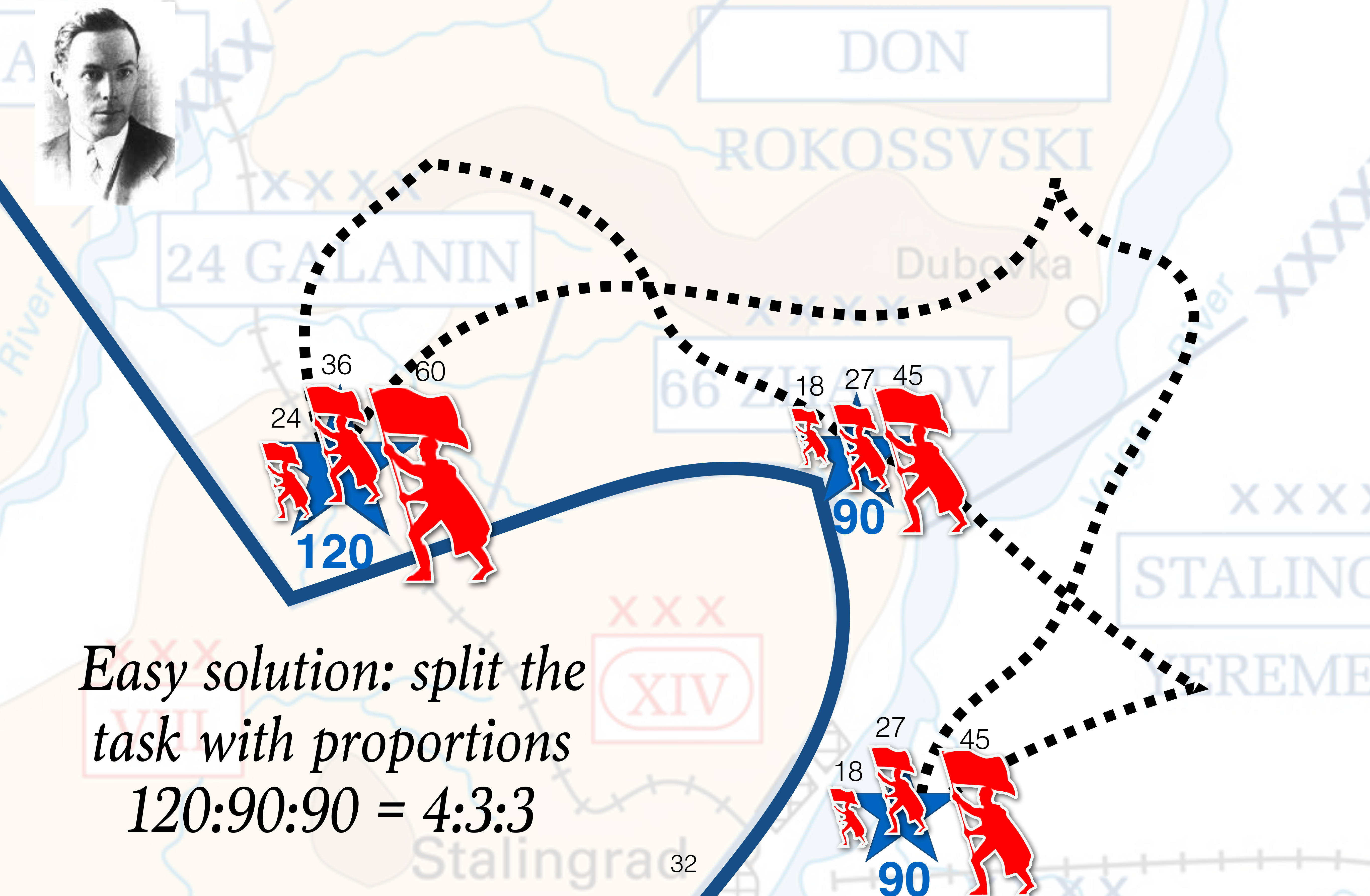
Easy solution: split the task with proportions 120:90:90 = 4:3:3

Kantorovich Problem



Easy solution: split the task with proportions 120:90:90 = 4:3:3

Kantorovich Problem



Easy solution: split the task with proportions 120:90:90 = 4:3:3

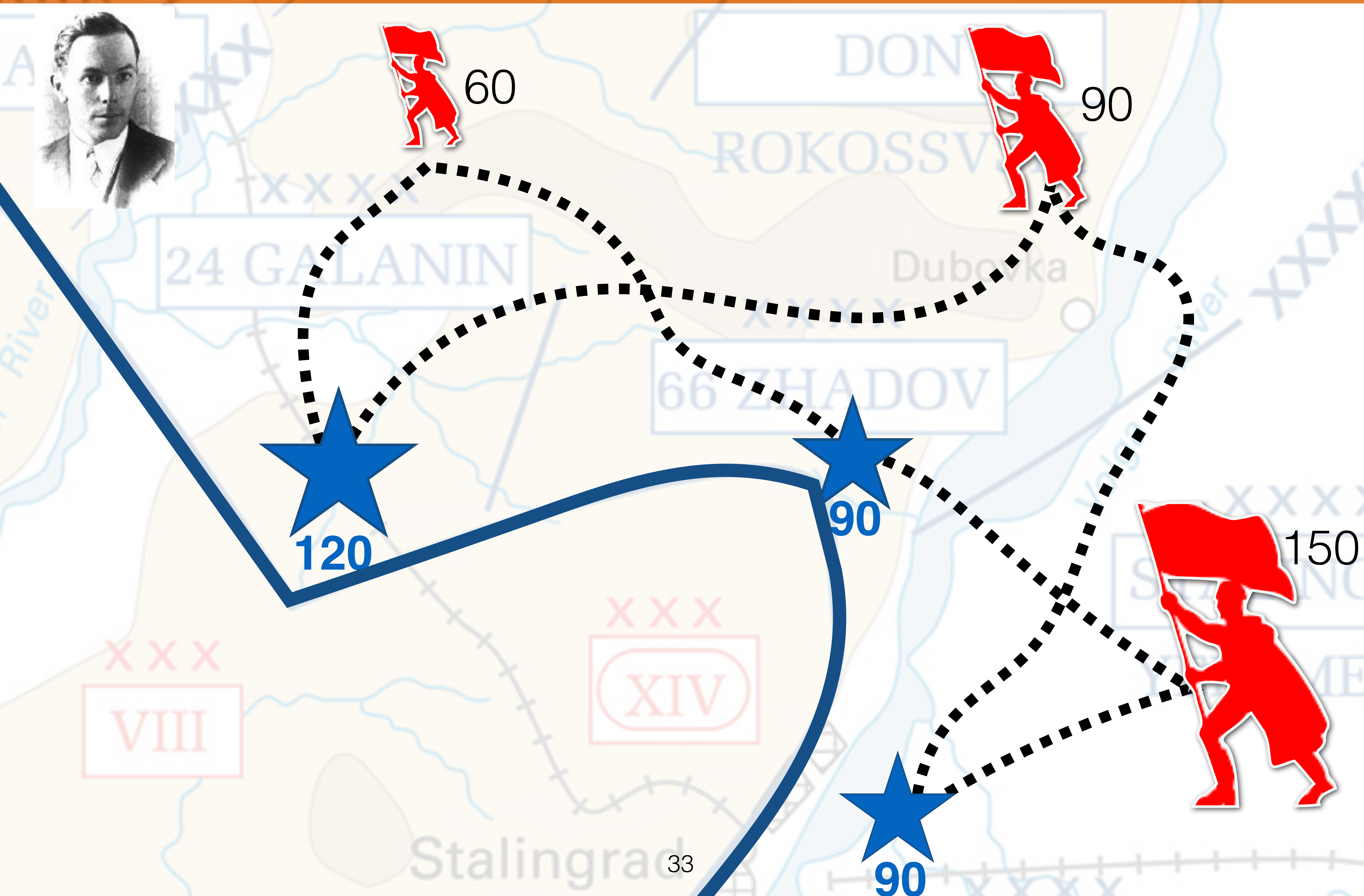
Kantorovich Problem

Naive approach results in many displacements…

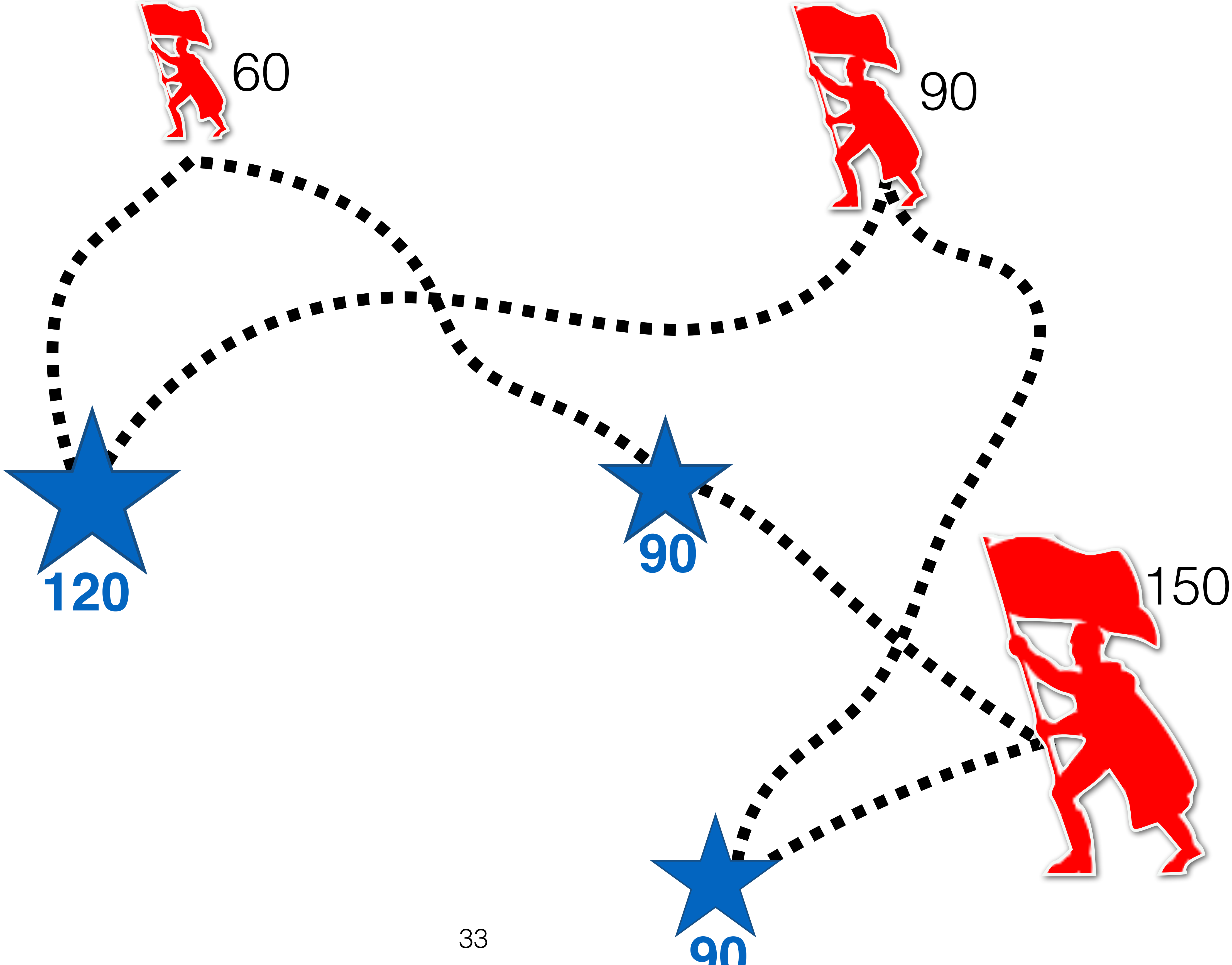
Can we find a cheaper alternative?

Easy solution: split the task with proportions 120:90:90 = 4:3:3

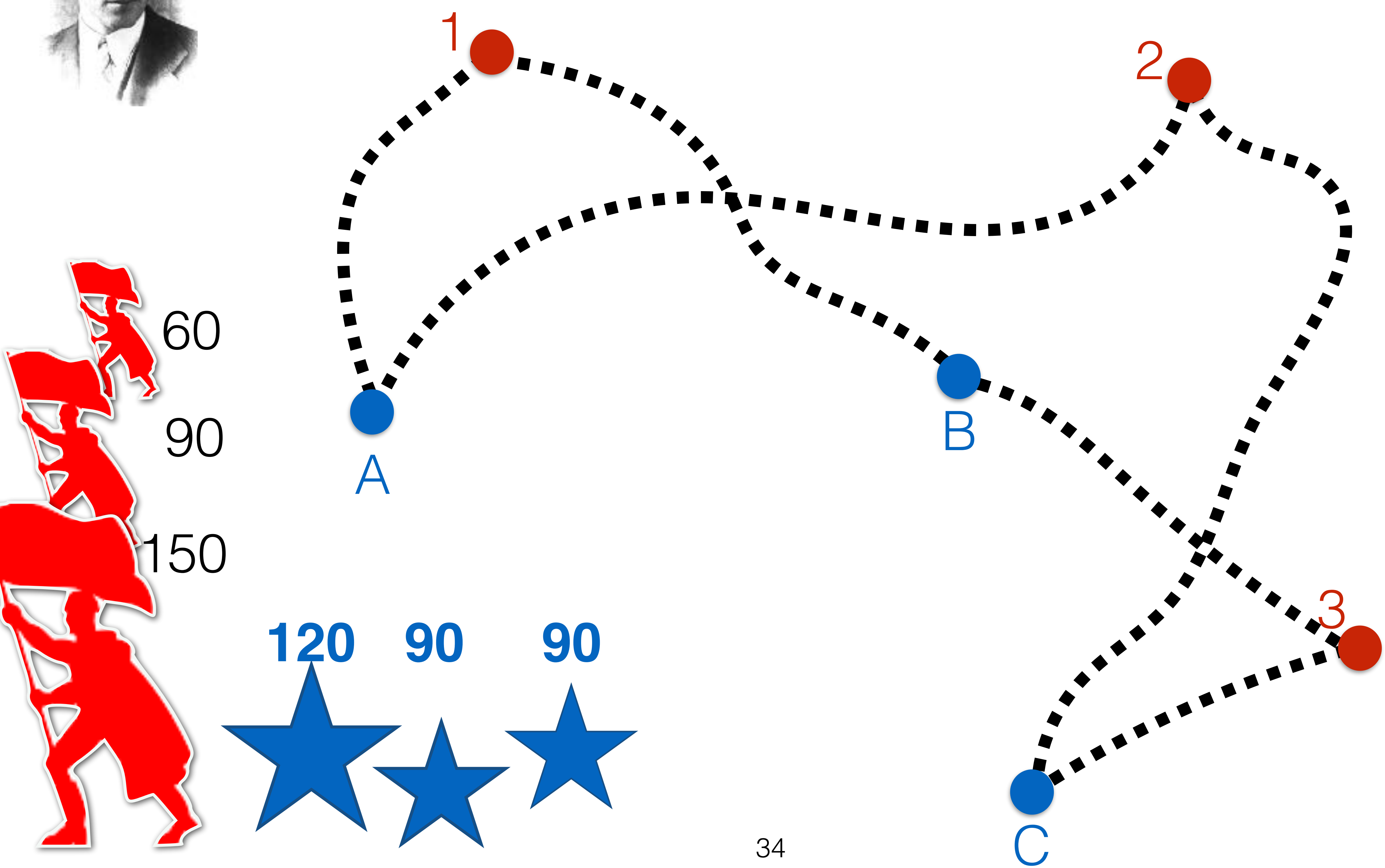
Kantorovich Problem



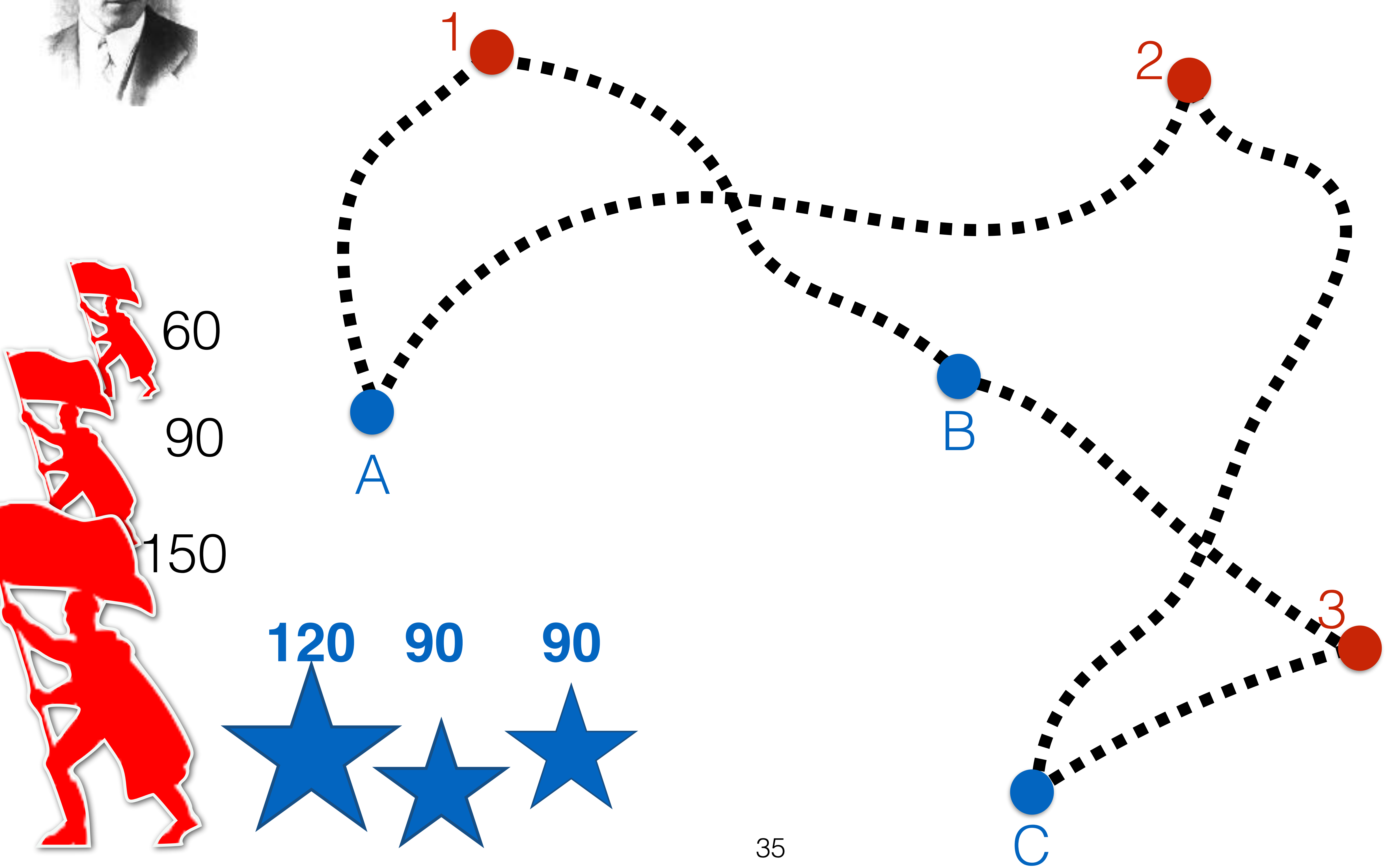
Kantorovich Problem



Kantorovich Problem



Kantorovich Problem



Kantorovich Problem

60	?	?	?
90	?	?	?
150	?	?	?
	120	90	90

1	$d_{\mathbf{1}\mathrm{A}}$	$d_{\mathbf{1}\mathrm{B}}$	$d_{\mathbf{1}\mathrm{C}}$
2	$d_{\mathbf{2}\mathrm{A}}$	$d_{\mathbf{2}\mathrm{B}}$	$d_{\mathbf{2}\mathrm{C}}$
3	$d_{\mathbf{3}\mathrm{A}}$	$d_{\mathbf{3}\mathrm{B}}$	$d_{\mathbf{3}\mathrm{C}}$
	A	B	C

Kantorovich Problem

Transportation matrix

	120	90	90
60	?	?	?
90	?	?	?
150	?	?	?

Distance matrix

	A	B	C
1	d_{1A}	d_{1B}	d_{1C}
2	d_{2A}	d_{2B}	d_{2C}
3	d_{3A}	d_{3B}	d_{3C}

Kantorovich Problem

The problem is entirely described by **counts** *and a* **cost/distance matrix**

Transportation matrix

60	?	?	?
90	?	?	?
150	?	?	?
	120	**90**	**90**

Distance matrix

	A	B	C
1	d_{1A}	d_{1B}	d_{1C}
2	d_{2A}	d_{2B}	d_{2C}
3	d_{3A}	d_{3B}	d_{3C}

Kantorovich Problem

Transportation matrix

	A	B	C
60	?	?	?
90	?	?	?
150	?	?	?
	120	**90**	**90**

Distance matrix

1	$d_{\mathbf{1}\mathrm{A}}$	$d_{\mathbf{1}\mathrm{B}}$	$d_{\mathbf{1}\mathrm{C}}$
2	$d_{\mathbf{2}\mathrm{A}}$	$d_{\mathbf{2}\mathrm{B}}$	$d_{\mathbf{2}\mathrm{C}}$
3	$d_{\mathbf{3}\mathrm{A}}$	$d_{\mathbf{3}\mathrm{B}}$	$d_{\mathbf{3}\mathrm{C}}$
	A	B	C

Kantorovich Problem

Transportation matrix

$$\begin{array}{c|c|c|c|}
 & & & \\
60 & p_{\mathbf{1}\mathrm{A}} & p_{\mathbf{1}\mathrm{B}} & p_{\mathbf{1}\mathrm{C}} \\
90 & p_{\mathbf{2}\mathrm{A}} & p_{\mathbf{2}\mathrm{B}} & p_{\mathbf{2}\mathrm{C}} \\
150 & p_{\mathbf{3}\mathrm{A}} & p_{\mathbf{3}\mathrm{B}} & p_{\mathbf{3}\mathrm{C}} \\
 & \mathbf{120} & \mathbf{90} & \mathbf{90}
\end{array}$$

Distance matrix

$$\begin{array}{c|c|c|c|}
1 & d_{\mathbf{1}\mathrm{A}} & d_{\mathbf{1}\mathrm{B}} & d_{\mathbf{1}\mathrm{C}} \\
2 & d_{\mathbf{2}\mathrm{A}} & d_{\mathbf{2}\mathrm{B}} & d_{\mathbf{2}\mathrm{C}} \\
3 & d_{\mathbf{3}\mathrm{A}} & d_{\mathbf{3}\mathrm{B}} & d_{\mathbf{3}\mathrm{C}} \\
 & \mathrm{A} & \mathrm{B} & \mathrm{C}
\end{array}$$

Kantorovich Problem

Transportation matrix

a_1	$p_{1\mathrm{A}}$	$p_{1\mathrm{B}}$	$p_{1\mathrm{C}}$
a_2	$p_{2\mathrm{A}}$	$p_{2\mathrm{B}}$	$p_{2\mathrm{C}}$
a_3	$p_{3\mathrm{A}}$	$p_{3\mathrm{B}}$	$p_{3\mathrm{C}}$
	b_{A}	b_{B}	b_{C}

Distance matrix

1	$d_{1\mathrm{A}}$	$d_{1\mathrm{B}}$	$d_{1\mathrm{C}}$
2	$d_{2\mathrm{A}}$	$d_{2\mathrm{B}}$	$d_{2\mathrm{C}}$
3	$d_{3\mathrm{A}}$	$d_{3\mathrm{B}}$	$d_{3\mathrm{C}}$
	A	B	C

Kantorovich Problem

Transportation matrix

a_1	p_{1A}	p_{1B}	p_{1C}
a_2	p_{2A}	p_{2B}	p_{2C}
a_3	p_{3A}	p_{3B}	p_{3C}
	b_A	b_B	b_C

Distance matrix

1	d_{1A}	d_{1B}	d_{1C}
2	d_{2A}	d_{2B}	d_{2C}
3	d_{3A}	d_{3B}	d_{3C}
	A	B	C

Constraints

$$\forall i \in \{1,2,3\}, \sum_{j \in \{A,B,C\}} p_{ij} = a_i$$

$$\forall j \in \{A,B,C\}, \sum_{i \in \{1,2,3\}} p_{ij} = b_j$$

$$p_{ij} \geq 0$$

Kantorovich Problem

Transportation matrix

a_1	$p_{1\mathrm{A}}$	$p_{1\mathrm{B}}$	$p_{1\mathrm{C}}$
a_2	$p_{2\mathrm{A}}$	$p_{2\mathrm{B}}$	$p_{2\mathrm{C}}$
a_3	$p_{3\mathrm{A}}$	$p_{3\mathrm{B}}$	$p_{3\mathrm{C}}$
	b_A	b_B	b_C

Distance matrix

1	$d_{1\mathrm{A}}$	$d_{1\mathrm{B}}$	$d_{1\mathrm{C}}$
2	$d_{2\mathrm{A}}$	$d_{2\mathrm{B}}$	$d_{2\mathrm{C}}$
3	$d_{3\mathrm{A}}$	$d_{3\mathrm{B}}$	$d_{3\mathrm{C}}$
	A	B	C

Constraints

$$\forall i \in \{1,2,3\}, \sum_{j\in\{\mathrm{A,B,C}\}} p_{ij} = a_i$$

$$\forall j \in \{\mathrm{A,B,C}\}, \sum_{i\in\{1,2,3\}} p_{ij} = b_j$$

$$p_{ij} \geq 0$$

Cost function

$$C(P) = \sum_{j\in\{\mathrm{A,B,C}\}} \sum_{i\in\{1,2,3\}} p_{ij} d_{ij}$$

Kantorovich Problem

Transportation matrix

$$\begin{array}{c|c|c|c|}
 & & & \\
a_1 & p_{1A} & p_{1B} & p_{1C} \\
a_2 & p_{2A} & p_{2B} & p_{2C} \\
a_3 & p_{3A} & p_{3B} & p_{3C} \\
 & b_A & b_B & b_C
\end{array}$$

Distance matrix

$$\begin{array}{c|c|c|c|}
 & & & \\
1 & d_{1A} & d_{1B} & d_{1C} \\
2 & d_{2A} & d_{2B} & d_{2C} \\
3 & d_{3A} & d_{3B} & d_{3C} \\
 & A & B & C
\end{array}$$

Constraints

$$\forall i \in \{1,2,3\}, \sum_{j \in \{A,B,C\}} p_{ij} = a_i$$

$$\forall j \in \{A,B,C\}, \sum_{i \in \{1,2,3\}} p_{ij} = b_j$$

$$p_{ij} \geq 0$$

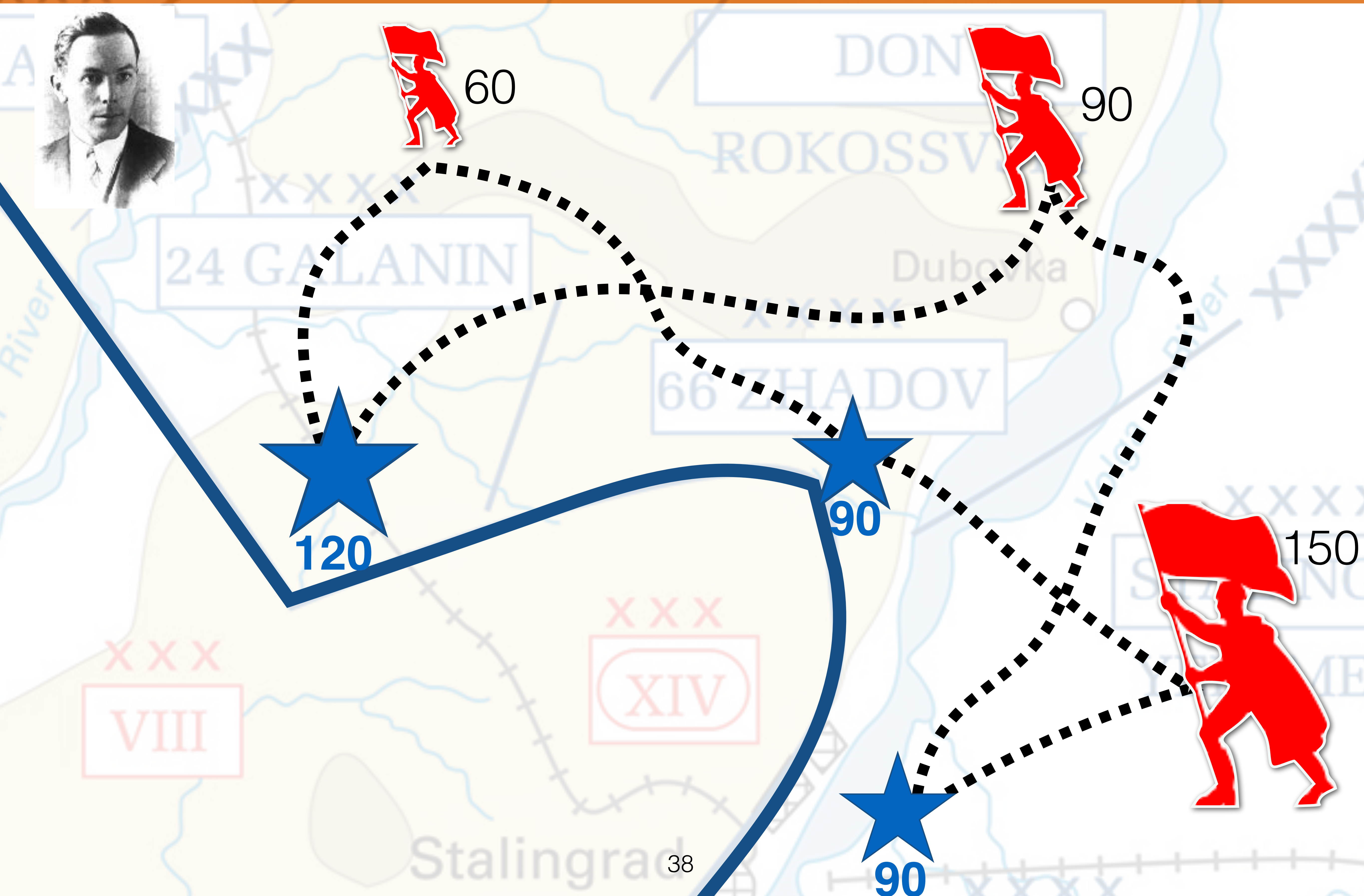
Cost function

$$C(P) = \sum_{j \in \{A,B,C\}} \sum_{i \in \{1,2,3\}} p_{ij} d_{ij}$$

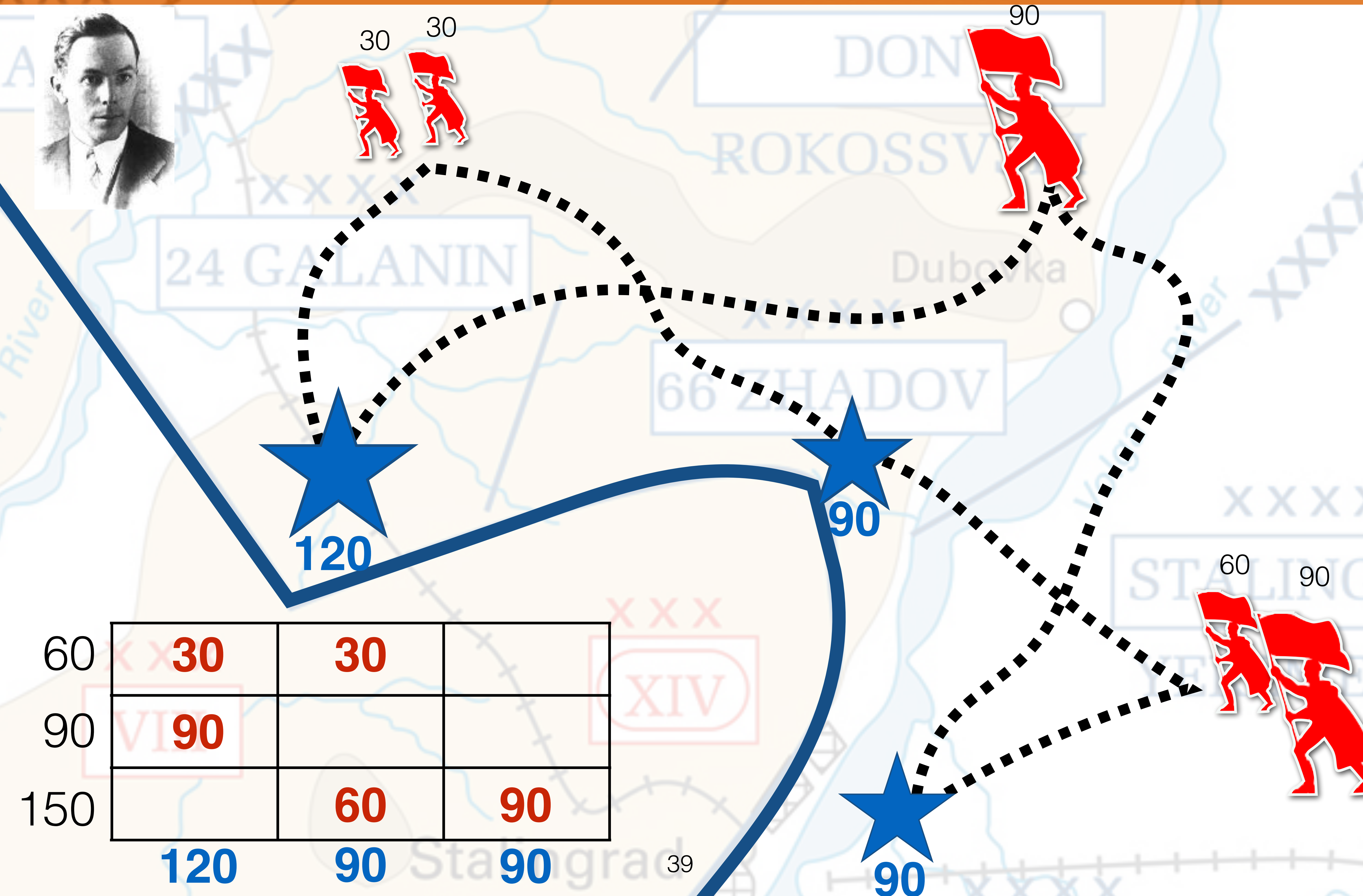
Problem

$$\min_{\text{all valid } P} C(P)$$

Kantorovich Problem

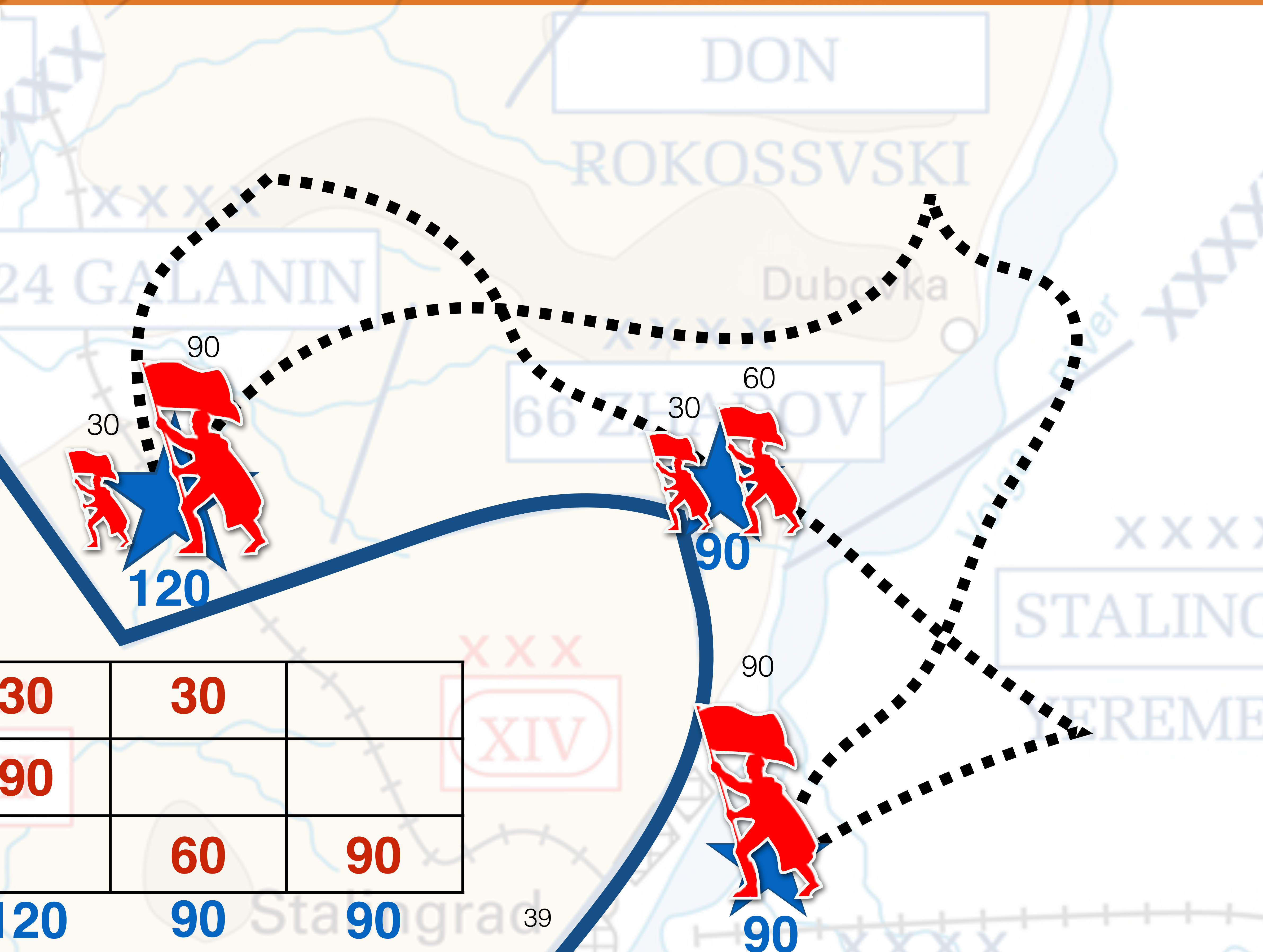


Kantorovich Problem



	120	90	90
60	30	30	
90	90		
150		60	90

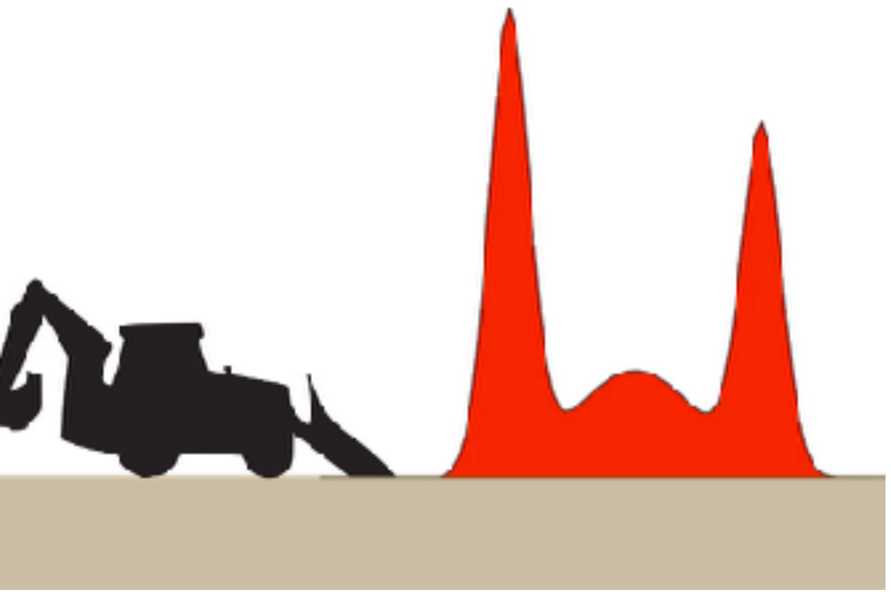
Kantorovich Problem

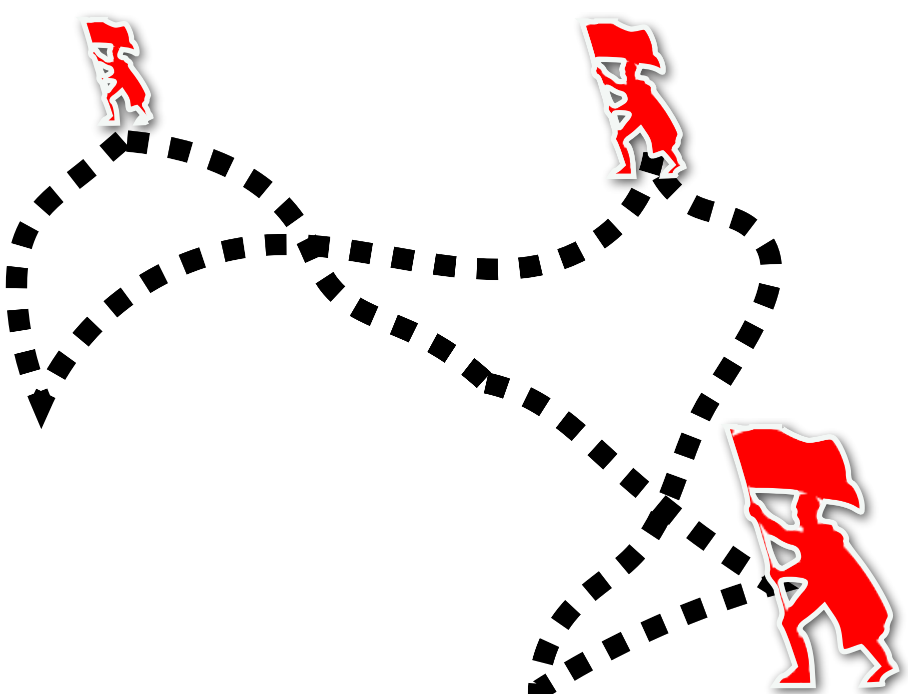


	120	90	90
60	30	30	
90	90		
150		60	90

Mathematical Formalism

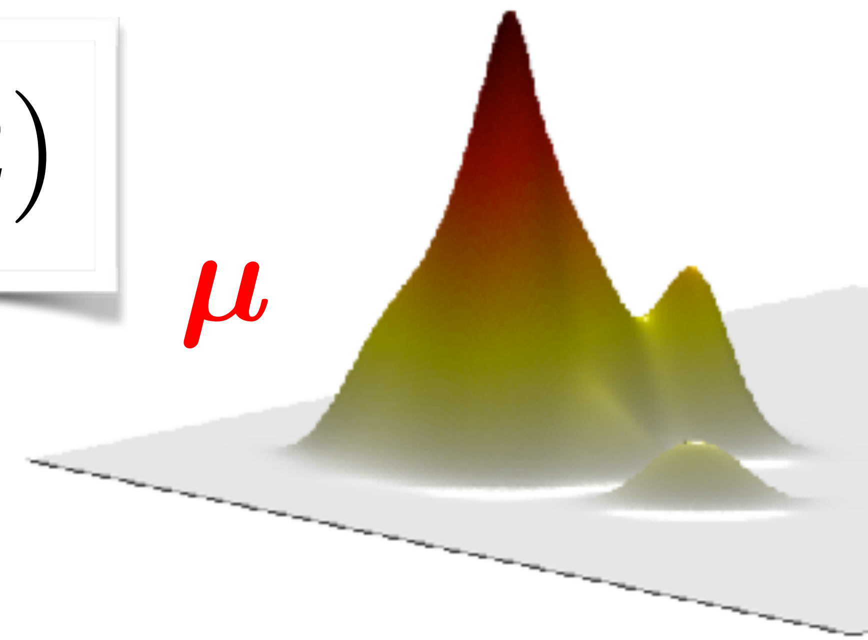
These problems involve discrete and continuous probability measures on a geometric space Ω





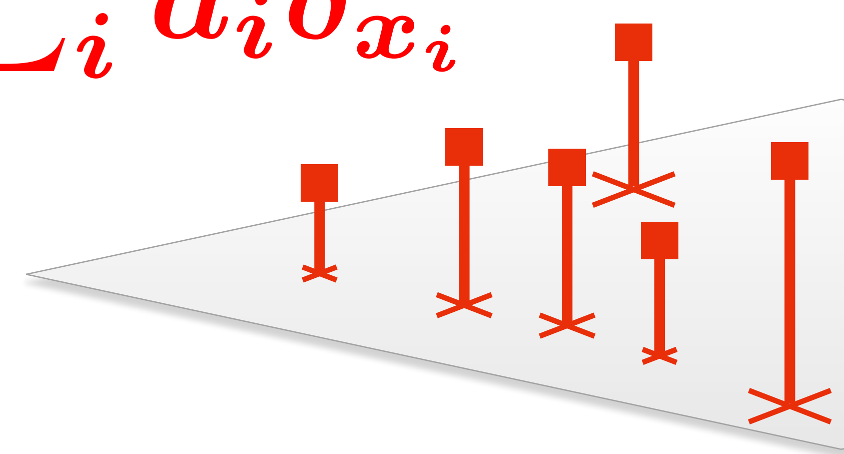
$\mathcal{P}(\Omega)$

μ



$\longmapsto$

$\sum_i a_i \delta_{x_i}$

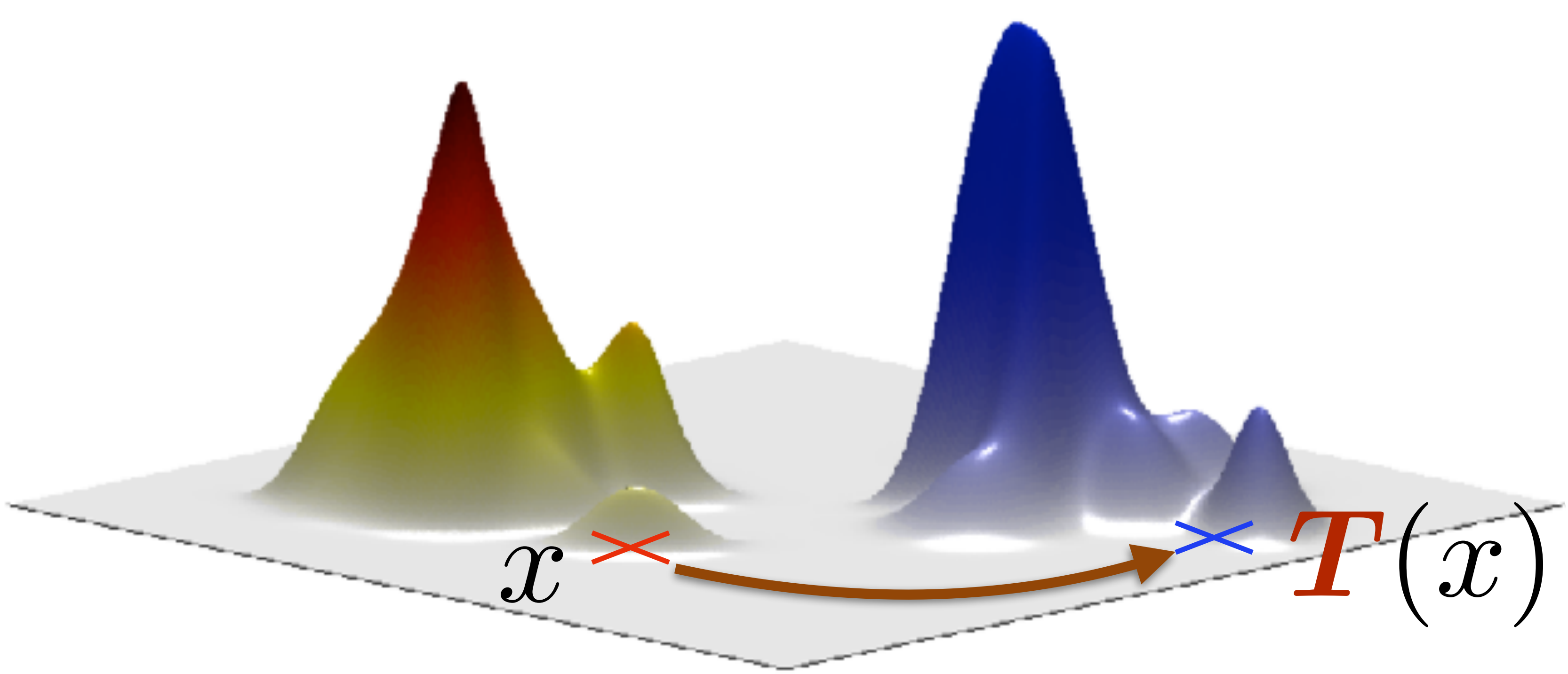


Monge Problem

Ω a measurable space, $\boldsymbol{c} : \Omega \times \Omega \to \mathbb{R}$.
$\boldsymbol{\mu}, \boldsymbol{\nu}$ two probability measures in $\mathcal{P}(\Omega)$.

[Monge'81] problem: find a map $\boldsymbol{T} : \Omega \to \Omega$

$$\inf_{\boldsymbol{T}_\sharp \boldsymbol{\mu} = \boldsymbol{\nu}} \int_\Omega \boldsymbol{c}(x, \boldsymbol{T}(x)) \boldsymbol{\mu}(dx)$$

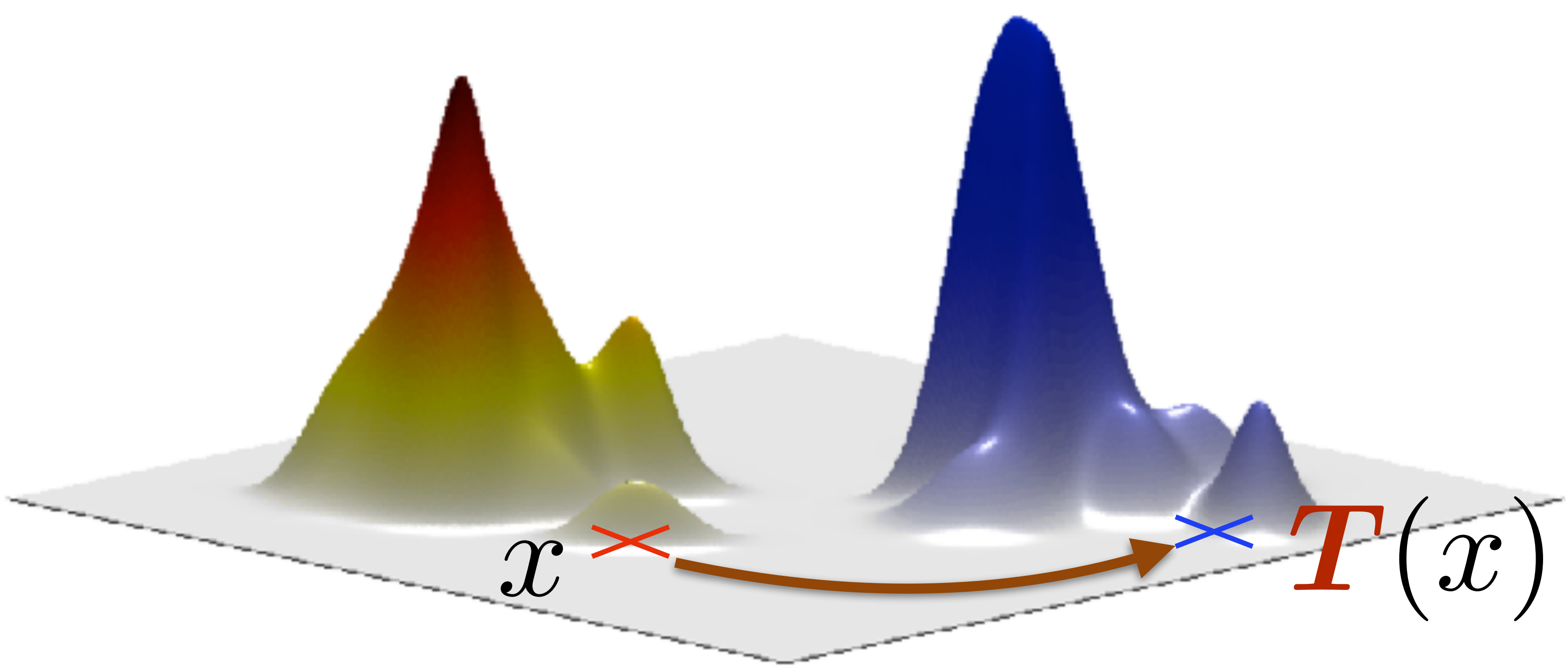


Monge Problem

Ω a measurable space, $\boldsymbol{c} : \Omega \times \Omega \to \mathbb{R}$.
$\boldsymbol{\mu}, \boldsymbol{\nu}$ two probability measures in $\mathcal{P}(\Omega)$.

[Monge'81] problem: find a map $\boldsymbol{T} : \Omega \to \Omega$

[Brenier'87] If $\Omega = \mathbb{R}^d$, $\boldsymbol{c} = \|\cdot - \cdot\|^2$,
$\boldsymbol{\mu}, \boldsymbol{\nu}$ a.c., then $\boldsymbol{T} = \nabla \boldsymbol{u}$, $\boldsymbol{u}$ **convex**.

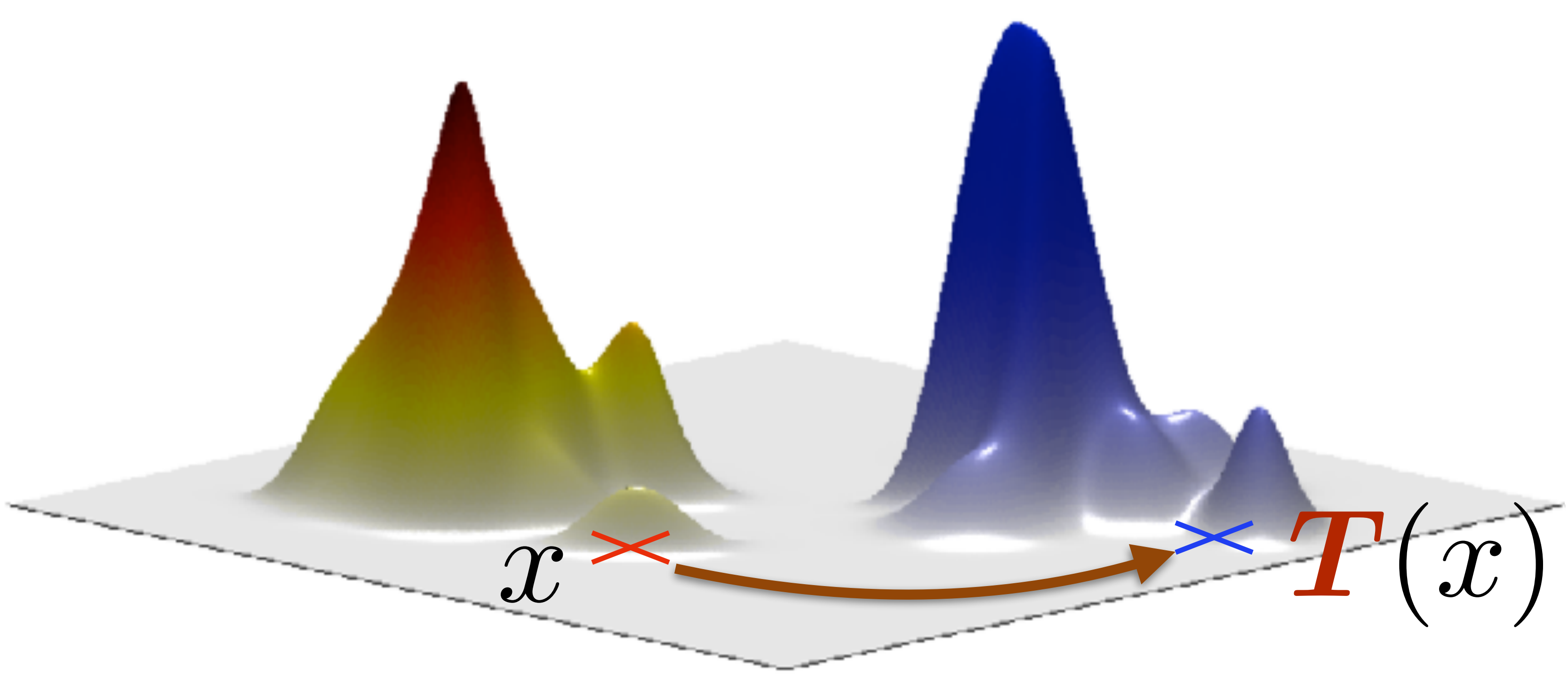


Monge Problem

Ω a measurable space, $c : \Omega \times \Omega \to \mathbb{R}$.
μ, ν two probability measures in $\mathcal{P}(\Omega)$.

[Monge'81] problem: find a map $T : \Omega \to \Omega$

$$\inf_{T_\sharp \mu = \nu} \int_\Omega c(x, T(x)) \mu(dx)$$

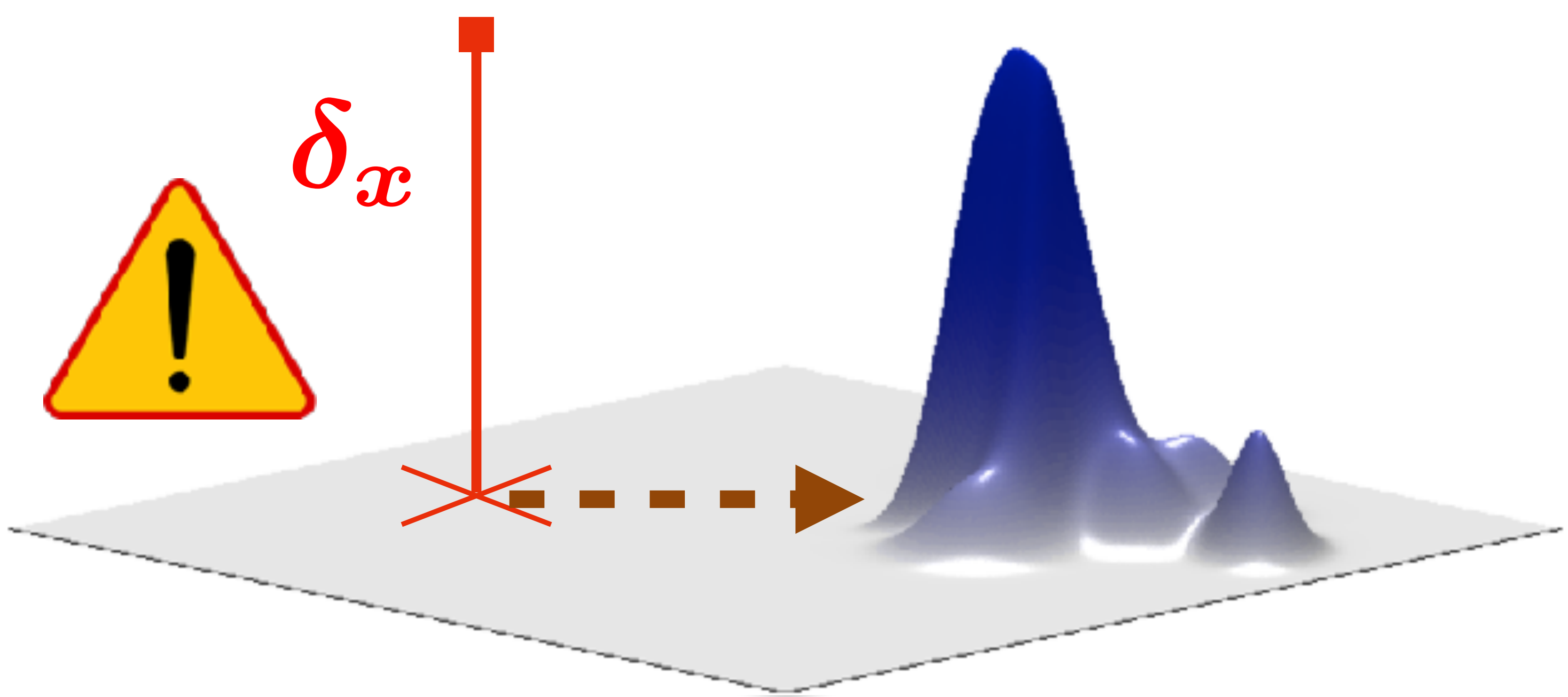


Monge Problem

Ω a measurable space, $\boldsymbol{c} : \Omega \times \Omega \to \mathbb{R}$.
$\boldsymbol{\mu}, \boldsymbol{\nu}$ two probability measures in $\mathcal{P}(\Omega)$.

[Monge'81] problem: find a map $\boldsymbol{T} : \Omega \to \Omega$

$$\inf_{\boldsymbol{T}_\sharp \boldsymbol{\mu} = \boldsymbol{\nu}} \int_\Omega \boldsymbol{c}(x, \boldsymbol{T}(x)) \boldsymbol{\mu}(dx)$$

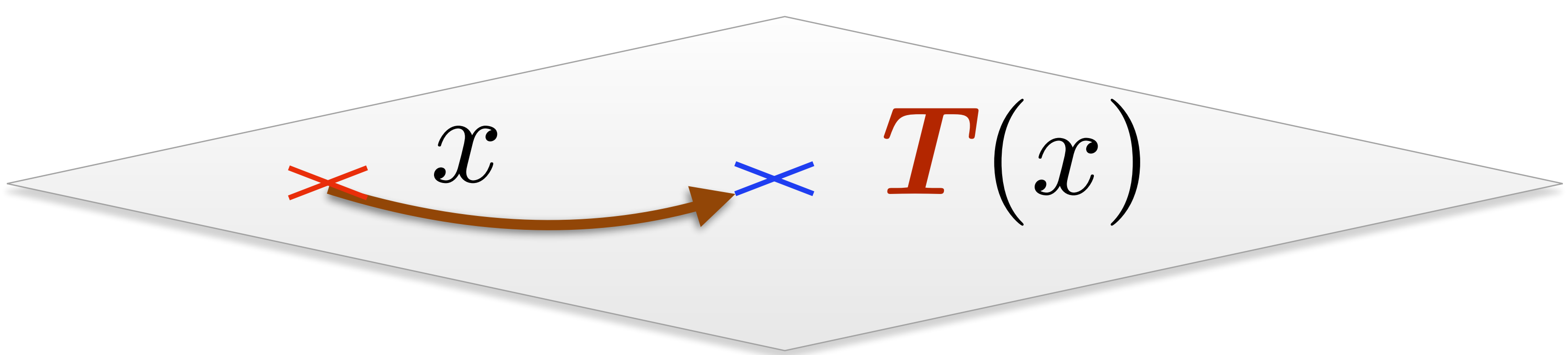


Kantorovich Relaxation

Instead of maps $\boldsymbol{T} : \Omega \to \Omega$,

consider probabilistic maps,

i.e. **couplings** $\boldsymbol{P} \in \mathcal{P}(\Omega \times \Omega)$:

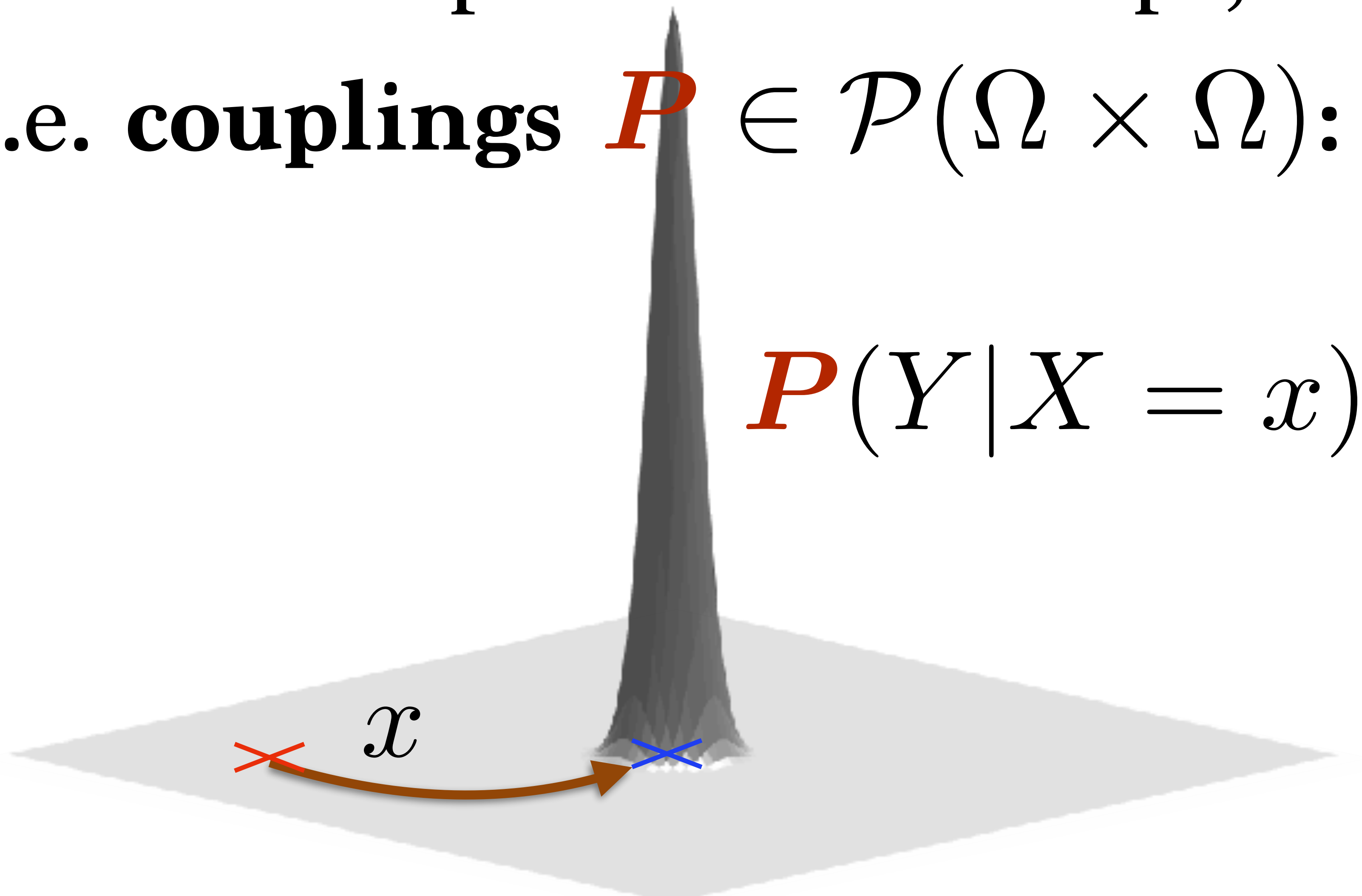


Kantorovich Relaxation

Instead of maps $\boldsymbol{T} : \Omega \to \Omega$,

consider probabilistic maps,

i.e. **couplings** $\boldsymbol{P} \in \mathcal{P}(\Omega \times \Omega)$:



Kantorovich Relaxation

Instead of maps $T : \Omega \to \Omega$,

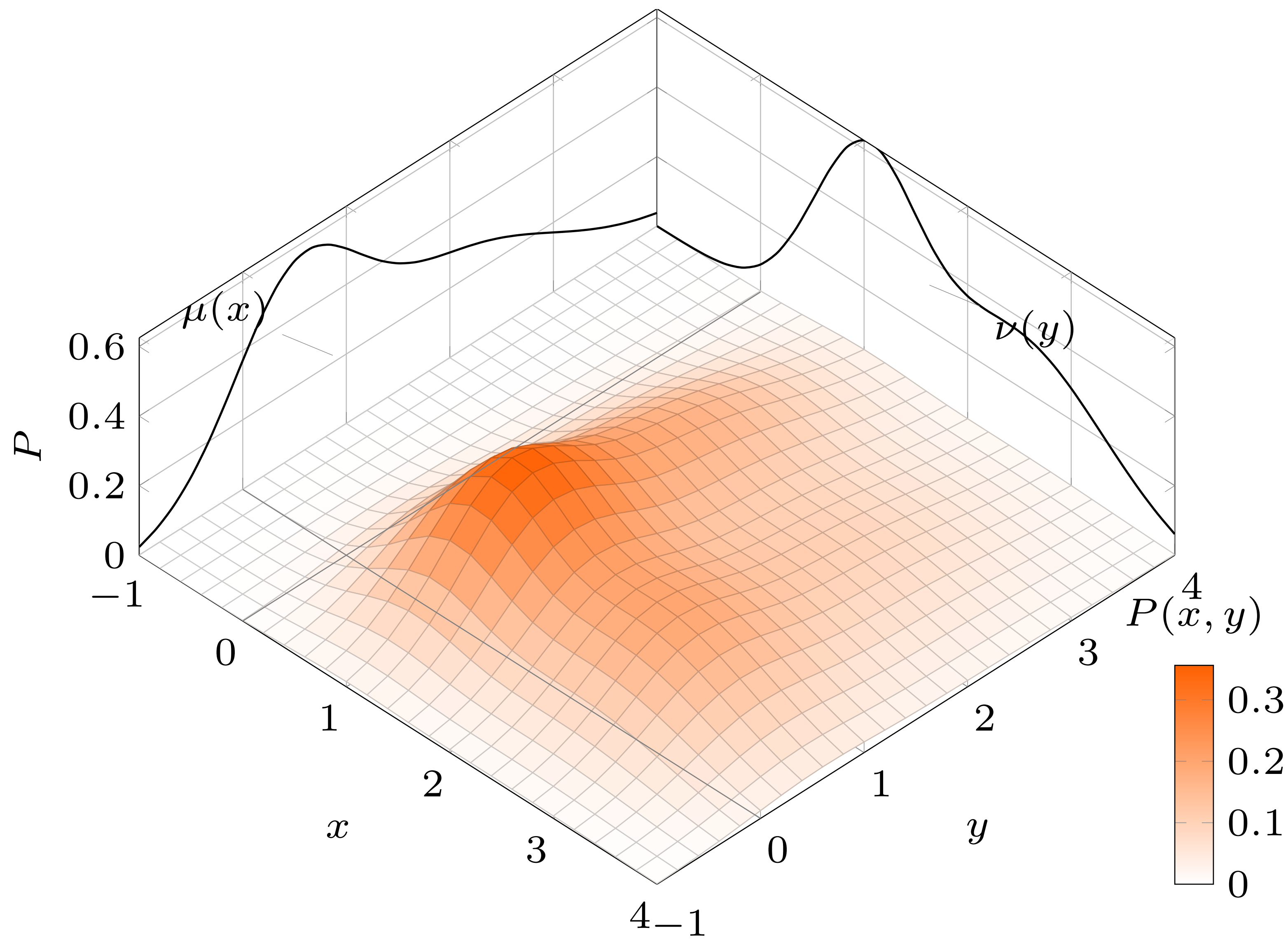
consider probabilistic maps,

i.e. **couplings** $P \in \mathcal{P}(\Omega \times \Omega)$:

$$\Pi(\mu, \nu) \overset{\text{def}}{=} \{P \in \mathcal{P}(\Omega \times \Omega) | \forall A, B \subset \Omega,$$
$$P(A \times \Omega) = \mu(A),$$
$$P(\Omega \times B) = \nu(B)\}$$

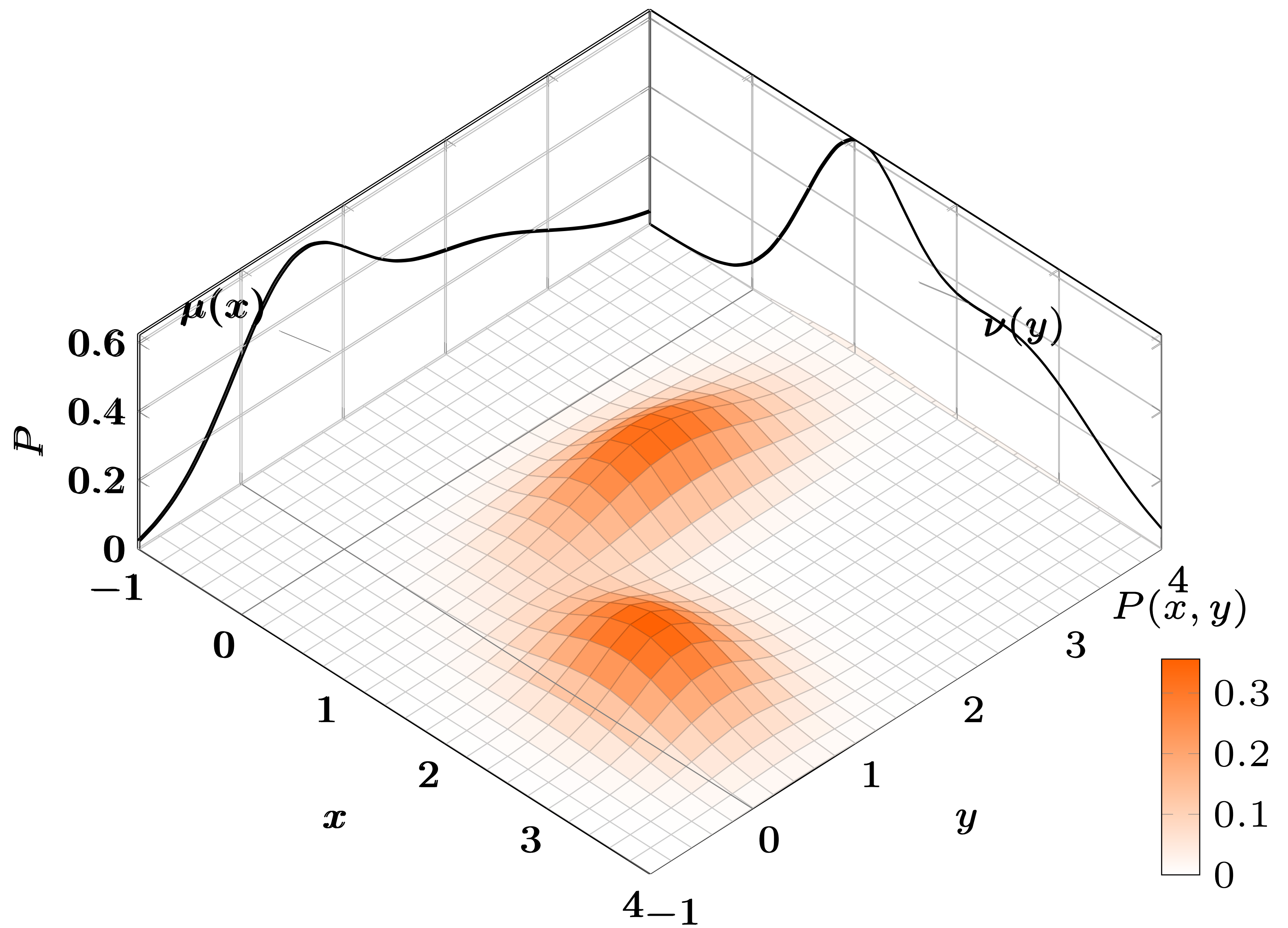
Kantorovich Relaxation

$$\Pi(\boldsymbol{\mu},\boldsymbol{\nu}) \stackrel{\text{def}}{=} \{\boldsymbol{P} \in \mathcal{P}(\Omega \times \Omega) | \forall \boldsymbol{A},\boldsymbol{B} \subset \Omega,$$
$$\boldsymbol{P}(\boldsymbol{A} \times \Omega) = \boldsymbol{\mu}(\boldsymbol{A}), \boldsymbol{P}(\Omega \times \boldsymbol{B}) = \boldsymbol{\nu}(\boldsymbol{B})\}$$



Kantorovich Relaxation

$$\Pi(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \{\boldsymbol{P} \in \mathcal{P}(\Omega \times \Omega) | \forall \boldsymbol{A}, \boldsymbol{B} \subset \Omega, \boldsymbol{P}(\boldsymbol{A} \times \Omega) = \boldsymbol{\mu}(\boldsymbol{A}), \boldsymbol{P}(\Omega \times \boldsymbol{B}) = \boldsymbol{\nu}(\boldsymbol{B})\}$$



Kantorovich Problem

$$\inf_{\boldsymbol{T}_\sharp \boldsymbol{\mu} = \boldsymbol{\nu}} \int_\Omega \boldsymbol{c}(x, \boldsymbol{T}(x)) \boldsymbol{\mu}(dx)$$

MONGE

Def. Given $\boldsymbol{\mu}, \boldsymbol{\nu}$ in $\mathcal{P}(\Omega)$; a cost function $\boldsymbol{c}$ on $\Omega \times \Omega$, the Kantorovich problem is

$$\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{c}(x, y) \boldsymbol{P}(dx, dy).$$

PRIMAL

Kantorovich Problem

Def. Given $\boldsymbol{\mu}, \boldsymbol{\nu}$ in $\mathcal{P}(\Omega)$; a cost function $\boldsymbol{c}$ on $\Omega \times \Omega$, the Kantorovich problem is

$$\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{c}(x, y) \boldsymbol{P}(dx, dy).$$

PRIMAL

Kantorovich Problem

Def. Given $\boldsymbol{\mu}, \boldsymbol{\nu}$ in $\mathcal{P}(\Omega)$; a cost function $\boldsymbol{c}$ on $\Omega \times \Omega$, the Kantorovich problem is

$$\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{c}(x, y) \boldsymbol{P}(dx, dy).$$

PRIMAL

$$\sup_{\substack{\boldsymbol{\varphi} \in L_1(\boldsymbol{\mu}), \boldsymbol{\psi} \in L_1(\boldsymbol{\nu}) \\ \boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y) \leq \boldsymbol{c}(x, y)}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu}.$$

DUAL

Kantorovich Problem

Def. Given $\boldsymbol{\mu}, \boldsymbol{\nu}$ in $\mathcal{P}(\Omega)$; a cost function $\boldsymbol{c}$ on $\Omega \times \Omega$, the Kantorovich problem is

$$\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{c}(x, y) \boldsymbol{P}(dx, dy).$$

PRIMAL

For two real-valued functions $\boldsymbol{\varphi}, \boldsymbol{\psi}$ on Ω,

$$(\boldsymbol{\varphi} \oplus \boldsymbol{\psi})(x, y) \stackrel{\text{def}}{=} \boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y)$$

Kantorovich Problem

Def. Given $\boldsymbol{\mu}, \boldsymbol{\nu}$ in $\mathcal{P}(\Omega)$; a cost function $\boldsymbol{c}$ on $\Omega \times \Omega$, the Kantorovich problem is

$$\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{c}(x, y) \boldsymbol{P}(dx, dy).$$

PRIMAL

$$\sup_{\substack{\boldsymbol{\varphi} \in L_1(\boldsymbol{\mu}), \boldsymbol{\psi} \in L_1(\boldsymbol{\nu}) \\ \boldsymbol{\varphi} \oplus \boldsymbol{\psi} \leq \boldsymbol{c}}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu}.$$

DUAL

Deriving Kantorovich Duality

$$\iota_\Pi(\boldsymbol{P}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \left[\int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iint \boldsymbol{\varphi} \oplus \boldsymbol{\psi} d\boldsymbol{P} \right]$$

$$= \begin{cases} 0 & \text{if } \boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu}), \\ +\infty & \text{otherwise.} \end{cases}$$

Deriving Kantorovich Duality

$$\iota_{\Pi}(\boldsymbol{P}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \left[\int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iint \boldsymbol{\varphi} \oplus \boldsymbol{\psi} d\boldsymbol{P}\right]$$

$$= \begin{cases} 0 & \text{if } \boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu}), \\ +\infty & \text{otherwise.} \end{cases}$$

$$\inf_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \iint \boldsymbol{c}\, d\boldsymbol{P}$$

Deriving Kantorovich Duality

$$\iota_\Pi(\boldsymbol{P}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \left[\int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iint \boldsymbol{\varphi} \oplus \boldsymbol{\psi} d\boldsymbol{P} \right]$$

$$= \begin{cases} 0 & \text{if } \boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu}), \\ +\infty & \text{otherwise.} \end{cases}$$

$$\inf_{\cancel{\boldsymbol{P} \in \Pi(\boldsymbol{\mu},\boldsymbol{\nu})}} \iint \boldsymbol{c} \, d\boldsymbol{P}$$

$$\inf_{\boldsymbol{P} \in \mathcal{P}_+(\Omega^2)} \iint \boldsymbol{c} \, d\boldsymbol{P} + \boxed{\iota_\Pi(\boldsymbol{P})}$$

Deriving Kantorovich Duality

$$\iota_\Pi(\boldsymbol{P}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \left[\int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iint \boldsymbol{\varphi} \oplus \boldsymbol{\psi} d\boldsymbol{P} \right]$$

$$= \begin{cases} 0 & \text{if } \boldsymbol{P} \in \Pi(\boldsymbol{\mu},\boldsymbol{\nu}), \\ +\infty & \text{otherwise.} \end{cases}$$

$$\boxed{\inf_{\boldsymbol{P}\in\mathcal{P}_+(\Omega^2)} \iint \boldsymbol{c}\, d\boldsymbol{P} + \iota_\Pi(\boldsymbol{P})}$$

Deriving Kantorovich Duality

$$\boxed{\inf_{P \in \mathcal{P}_+(\Omega^2)} \iint c \, dP + \iota_\Pi(P)}$$

$$\inf_{P \in \mathcal{P}_+(\Omega^2)} \iint c \, dP + \sup_{\varphi, \psi} \int \varphi d\mu + \int \psi d\nu - \iint \varphi \oplus \psi dP$$

Deriving Kantorovich Duality

$$\boxed{\inf_{\boldsymbol{P}\in\mathcal{P}_+(\Omega^2)} \iint \boldsymbol{c}\, d\boldsymbol{P} + \iota_\Pi(\boldsymbol{P})}$$

$$\inf_{\boldsymbol{P}\in\mathcal{P}_+(\Omega^2)} \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \iint \boldsymbol{c}\, d\boldsymbol{P} + \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iint \boldsymbol{\varphi} \oplus \boldsymbol{\psi} d\boldsymbol{P}$$

Deriving Kantorovich Duality

$$\boxed{\inf_{P \in \mathcal{P}_+(\Omega^2)} \iint c \, dP + \iota_\Pi(P)}$$

$$\inf_{P \in \mathcal{P}_+(\Omega^2)} \sup_{\varphi, \psi} \iint c \, dP - \iint \varphi \oplus \psi dP + \int \varphi d\mu + \int \psi d\nu$$

Deriving Kantorovich Duality

$$\boxed{\inf_{\boldsymbol{P} \in \mathcal{P}_+(\Omega^2)} \iint \boldsymbol{c}\, d\boldsymbol{P} + \iota_\Pi(\boldsymbol{P})}$$

$$\inf_{\boldsymbol{P} \in \mathcal{P}_+(\Omega^2)} \sup_{\boldsymbol{\varphi}, \boldsymbol{\psi}} \iint (\boldsymbol{c} - \boldsymbol{\varphi} \oplus \boldsymbol{\psi}) d\boldsymbol{P} + \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu}$$

Deriving Kantorovich Duality

$$\boxed{\inf_{\boldsymbol{P}\in\mathcal{P}_+(\Omega^2)} \iint \boldsymbol{c}\, d\boldsymbol{P} + \iota_\Pi(\boldsymbol{P})}$$

$$\sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \inf_{\boldsymbol{P}\in\mathcal{P}_+(\Omega^2)} \iint (\boldsymbol{c} - \boldsymbol{\varphi} \oplus \boldsymbol{\psi}) d\boldsymbol{P} \qquad + \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu}$$

Deriving Kantorovich Duality

$$\boxed{\inf_{P \in \mathcal{P}_+(\Omega^2)} \iint c\, dP + \iota_\Pi(P)}$$

$$\sup_{\varphi,\psi} \inf_{P \in \mathcal{P}_+(\Omega^2)} \iint (c - \varphi \oplus \psi) dP \qquad + \int \varphi d\mu + \int \psi d\nu$$

Deriving Kantorovich Duality

$$\boxed{\inf_{P \in \mathcal{P}_+(\Omega^2)} \iint c \, dP + \iota_\Pi(P)}$$

$$\sup_{\varphi, \psi} \inf_{P \in \mathcal{P}_+(\Omega^2)} \iint (c - \varphi \oplus \psi) dP \qquad + \int \varphi d\mu + \int \psi d\nu$$

$$\inf_{P \in \mathcal{P}_+(\Omega)} \iint (c - \varphi \oplus \psi) dP = \begin{cases} 0 & \text{if } c - \varphi \oplus \psi \geq 0. \\ -\infty & \text{otherwise} \end{cases}$$

Deriving Kantorovich Duality

$$\boxed{\inf_{P\in\mathcal{P}_+(\Omega^2)} \iint c\, dP + \iota_\Pi(P)}$$

$$\sup_{\varphi,\psi} \inf_{P\in\mathcal{P}_+(\Omega^2)} \iint (c - \varphi\oplus\psi)dP \qquad + \int \varphi d\mu + \int \psi d\nu$$

$$\inf_{P\in\mathcal{P}_+(\Omega)} \iint (c - \varphi\oplus\psi)dP = \begin{cases} 0 & \text{if } c - \varphi\oplus\psi \geq 0. \\ -\infty & \text{otherwise} \end{cases}$$

$$\boxed{\sup_{\varphi\oplus\psi\leq c} \int \varphi d\mu + \int \psi d\nu.}$$

DUAL

Wasserstein Distances

Let $p \geq 1$. Let $\boldsymbol{c}(x, y) := \boldsymbol{D}^p(x, y)$, a metric.

Def. The p-**Wasserstein distance** between $\boldsymbol{\mu}, \boldsymbol{\nu}$ in $\mathcal{P}(\Omega)$ is

$$W_p(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \left(\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{D}(\boldsymbol{x}, \boldsymbol{y})^p \boldsymbol{P}(dx, dy) \right)^{1/p}.$$

Wasserstein Distances

Let $p \geq 1$. Let $\boldsymbol{c}(x,y) := \boldsymbol{D}^p(x,y)$, a metric.

Def. The p-**Wasserstein distance** between $\boldsymbol{\mu}, \boldsymbol{\nu}$ in $\mathcal{P}(\Omega)$ is

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) \overset{\text{def}}{=} \left(\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{D}(\boldsymbol{x}, \boldsymbol{y})^p \boldsymbol{P}(dx, dy) \right)^{\cancel{1/p}}.$$

Kantorovich Duality

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\substack{\boldsymbol{\varphi} \in L_1(\boldsymbol{\mu}), \boldsymbol{\psi} \in L_1(\boldsymbol{\nu}) \\ \boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y) \leq \boldsymbol{D}^p(x,y)}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu}.$$

DUAL

- Kantorovich Duality is **interesting** from a computational perspective: easier to store 2 functions than a whole coupling.
- D transforms: go from **two** to **one** dual potential.

D transforms

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\substack{\boldsymbol{\varphi} \in L_1(\boldsymbol{\mu}), \boldsymbol{\psi} \in L_1(\boldsymbol{\nu}) \\ \boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y) \leq \boldsymbol{D}^p(x,y)}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu}.$$

DUAL

Imagine we choose a $\boldsymbol{\varphi}$. Can we find a good $\boldsymbol{\psi}$?

D transforms

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\substack{\varphi \in L_1(\boldsymbol{\mu}), \psi \in L_1(\boldsymbol{\nu}) \\ \varphi(x) + \psi(y) \leq \boldsymbol{D}^p(x,y)}} \int \varphi \, d\boldsymbol{\mu} + \int \psi \, d\boldsymbol{\nu}.$$

DUAL

Imagine we choose a φ. Can we find a good ψ?

We need that ψ satisfies for *all* $\boldsymbol{x}, \boldsymbol{y}$

$$\varphi(\boldsymbol{x}) + \psi(\boldsymbol{y}) \leq \boldsymbol{D}^p(\boldsymbol{x}, \boldsymbol{y})$$

D transforms

$$W_p^p(\mu,\nu) = \sup_{\substack{\varphi\in L_1(\mu),\psi\in L_1(\nu)\\ \varphi(x)+\psi(y)\leq D^p(x,y)}} \int \varphi d\mu + \int \psi d\nu.$$

DUAL

Imagine we choose a φ. Can we find a good ψ?

We need that ψ satisfies for *all* x, y

$$\varphi(x) + \psi(y) \leq D^p(x,y)$$

$$\psi(y) \leq D^p(x,y) - \varphi(x)$$

D transforms

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\substack{\varphi \in L_1(\boldsymbol{\mu}), \psi \in L_1(\boldsymbol{\nu}) \\ \varphi(x) + \psi(y) \leq \boldsymbol{D}^p(x,y)}} \int \varphi \, d\boldsymbol{\mu} + \int \psi \, d\boldsymbol{\nu}.$$

DUAL

Imagine we choose a φ. Can we find a good ψ?

We need that ψ satisfies for *all* $\boldsymbol{x}, \boldsymbol{y}$

$$\varphi(\boldsymbol{x}) + \psi(\boldsymbol{y}) \leq \boldsymbol{D}^p(\boldsymbol{x}, \boldsymbol{y})$$

$$\psi(\boldsymbol{y}) \leq \boldsymbol{D}^p(\boldsymbol{x}, \boldsymbol{y}) - \varphi(\boldsymbol{x})$$

$$\psi(\boldsymbol{y}) \leq \inf_{\boldsymbol{x}} \boldsymbol{D}^p(\boldsymbol{x}, \boldsymbol{y}) - \varphi(\boldsymbol{x})$$

D transforms

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\substack{\varphi\in L_1(\boldsymbol{\mu}),\psi\in L_1(\boldsymbol{\nu})\\ \varphi(x)+\psi(y)\leq \boldsymbol{D}^p(x,y)}} \int \varphi d\boldsymbol{\mu} + \int \psi d\boldsymbol{\nu}.$$

DUAL

For given φ, cannot get a better ψ than

$$\overline{\varphi}(\boldsymbol{y}) \stackrel{\text{def}}{=} \inf_{\boldsymbol{x}} \boldsymbol{D}^p(\boldsymbol{x},\boldsymbol{y}) - \varphi(\boldsymbol{x}).$$

D transforms

$$W_p^p(\mu,\nu) = \sup_{\substack{\varphi\in L_1(\mu),\psi\in L_1(\nu)\\ \varphi(x)+\psi(y)\leq D^p(x,y)}} \int \varphi d\mu + \int \psi d\nu.$$

DUAL

For given φ, cannot get a better ψ than

$$\overline{\varphi}(y) \stackrel{\text{def}}{=} \inf_{x} D^p(x,y) - \varphi(x).$$

$$W_p^p(\mu,\nu) = \sup_{\varphi} \int \varphi d\mu + \int \overline{\varphi} d\nu.$$

SEMI-DUAL

D transforms

$$\overline{\varphi}(\boldsymbol{y}) \stackrel{\text{def}}{=} \inf_{\boldsymbol{x}} D^p(\boldsymbol{x}, \boldsymbol{y}) - \varphi(\boldsymbol{x}).$$

$$\overline{\psi}(\boldsymbol{x}) = \inf_{\boldsymbol{y}} D^p(\boldsymbol{x}, \boldsymbol{y}) - \psi(\boldsymbol{y}).$$

D transforms

$$\overline{\varphi}(y) \stackrel{\text{def}}{=} \inf_{x} D^p(x, y) - \varphi(x).$$

$$\overline{\psi}(x) = \inf_{y} D^p(x, y) - \psi(y).$$

$$W_p^p(\mu, \nu) = \sup_{\varphi} \int \overline{\overline{\varphi}} d\mu + \int \overline{\varphi} d\nu.$$

D transforms

$$\overline{\varphi}(y) \stackrel{\text{def}}{=} \inf_{x} D^p(x,y) - \varphi(x).$$

$$\overline{\psi}(x) = \inf_{y} D^p(x,y) - \psi(y).$$

$$W_p^p(\mu,\nu) = \sup_{\varphi} \int \overline{\overline{\varphi}}\, d\mu + \int \overline{\varphi}\, d\nu.$$

$$\overline{\overline{\overline{\overline{\overline{\overline{\varphi}}}}}}\ ??$$

D transforms

$$\overline{\varphi}(\boldsymbol{y}) \stackrel{\text{def}}{=} \inf_{\boldsymbol{x}} D^p(\boldsymbol{x}, \boldsymbol{y}) - \varphi(\boldsymbol{x}).$$

$$\overline{\psi}(\boldsymbol{x}) = \inf_{\boldsymbol{y}} D^p(\boldsymbol{x}, \boldsymbol{y}) - \psi(\boldsymbol{y}).$$

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\varphi} \int \overline{\overline{\varphi}} d\boldsymbol{\mu} + \int \overline{\varphi} d\boldsymbol{\nu}.$$

For all φ, we have $\overline{\overline{\overline{\varphi}}} = \overline{\varphi}$

D transforms

$$\overline{\varphi}(\boldsymbol{y}) \stackrel{\text{def}}{=} \inf_{\boldsymbol{x}} D^p(\boldsymbol{x},\boldsymbol{y}) - \varphi(\boldsymbol{x}).$$

$$\overline{\psi}(\boldsymbol{x}) = \inf_{\boldsymbol{y}} D^p(\boldsymbol{x},\boldsymbol{y}) - \psi(\boldsymbol{y}).$$

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\varphi} \int \overline{\overline{\varphi}}\, d\boldsymbol{\mu} + \int \overline{\varphi}\, d\boldsymbol{\nu}.$$

For all φ, we have $\overline{\overline{\overline{\varphi}}} = \overline{\varphi}$

φ is D^p-concave if $\exists \phi : \varphi = \overline{\phi}$

φ is D^p-concave $\Rightarrow \overline{\overline{\varphi}} = \varphi$

D transforms

$$\overline{\varphi}(y) \stackrel{\text{def}}{=} \inf_{x} D^p(x, y) - \varphi(x).$$

$$\overline{\psi}(x) = \inf_{y} D^p(x, y) - \psi(y).$$

$$W_p^p(\mu, \nu) = \sup_{\varphi} \int \overline{\overline{\varphi}} d\mu + \int \overline{\varphi} d\nu.$$

$$W_p^p(\mu, \nu) = \sup_{\varphi \text{ is } D^p\text{-concave}} \int \varphi d\mu + \int \overline{\varphi} d\nu.$$

D transforms, W_1

Prop. If $\boldsymbol{c} = \boldsymbol{D}$, namely $p = 1$, then φ is $\boldsymbol{D}$-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

For given $\boldsymbol{x}$, $\overline{\varphi}_{\boldsymbol{x}}(\boldsymbol{y}) \stackrel{\text{def}}{=} \boldsymbol{D}(\boldsymbol{x}, \boldsymbol{y}) - \varphi(\boldsymbol{x})$ is 1-Lipschitz.

D transforms, W_1

Prop. If $c = D$, namely $p = 1$, then
φ is D-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

For given x, $\overline{\varphi}_x(y) \stackrel{\text{def}}{=} D(x,y) - \varphi(x)$ is 1-Lipschitz.

$$\overline{\varphi}_x(y) - \overline{\varphi}_x(y') = D(x,y) - D(x,y') \leq D(y,y')$$

D transforms, W_1

Prop. If $c = D$, namely $p = 1$, then
φ is D-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

For given x, $\overline{\varphi}_x(y) \stackrel{\text{def}}{=} D(x, y) - \varphi(x)$ is 1-Lipschitz.

D transforms, W_1

Prop. If $c = D$, namely $p = 1$, then φ is D-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

For given x, $\overline{\varphi}_x(y) \overset{\text{def}}{=} D(x, y) - \varphi(x)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) = \inf_x \overline{\varphi}_x(y)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) - \overline{\varphi}(x) \leq D(x, y)$

D transforms, W_1

Prop. If $c = D$, namely $p = 1$, then
φ is D-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

For given x, $\overline{\varphi}_x(y) \stackrel{\text{def}}{=} D(x, y) - \varphi(x)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) = \inf_x \overline{\varphi}_x(y)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) - \overline{\varphi}(x) \leq D(x, y)$

$\Rightarrow -\overline{\varphi}(x) \leq D(x, y) - \overline{\varphi}(y)$

$\Rightarrow -\overline{\varphi}(x) \leq \inf_y D(x, y) - \overline{\varphi}(y)$

D transforms, W_1

Prop. If $c = D$, namely $p = 1$, then
φ is D-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

For given x, $\overline{\varphi}_x(y) \stackrel{\text{def}}{=} D(x, y) - \varphi(x)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) = \inf_x \overline{\varphi}_x(y)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) - \overline{\varphi}(x) \leq D(x, y)$

$\Rightarrow -\overline{\varphi}(x) \leq D(x, y) - \overline{\varphi}(y)$

$\Rightarrow -\overline{\varphi}(x) \leq \inf_y D(x, y) - \overline{\varphi}(y)$

$\Rightarrow -\overline{\varphi}(x) \leq \inf_y D(x, y) - \overline{\varphi}(y) \leq -\overline{\varphi}(x)$

D transforms, W_1

Prop. If $c = D$, namely $p = 1$, then φ is D-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

For given x, $\overline{\varphi}_x(y) \overset{\text{def}}{=} D(x, y) - \varphi(x)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) = \inf_x \overline{\varphi}_x(y)$ is 1-Lipschitz.

$\Rightarrow \overline{\varphi}(y) - \overline{\varphi}(x) \leq D(x, y)$

$\Rightarrow -\overline{\varphi}(x) \leq D(x, y) - \overline{\varphi}(y)$

$\Rightarrow -\overline{\varphi}(x) \leq \inf_y D(x, y) - \overline{\varphi}(y)$

$\Rightarrow -\overline{\varphi}(x) \leq \inf_y D(x, y) - \overline{\varphi}(y) \leq -\overline{\varphi}(x)$

$\Rightarrow -\overline{\varphi}(x) \leq \overline{\overline{\varphi}}(x) \leq -\overline{\varphi}(x)$ and $\overline{\varphi}(x) = -\varphi(x)$

D transforms, W_1

$$W_1(\mu,\nu) = \sup_{\varphi \text{ is } D\text{-concave}} \int \varphi\, d\mu + \int \overline{\varphi}\, d\nu.$$

SEMI-DUAL

Prop. If $c = D$, then
φ is D-concave $\Leftrightarrow \overline{\varphi} = -\varphi$, φ is 1-Lipschitz

$$W_1(\mu,\nu) = \sup_{\varphi \text{ 1-Lipschitz}} \int \varphi(d\mu - d\nu).$$

W1

Links between Monge & Kantorovich

Prop. For "well behaved" costs $\boldsymbol{c}$, if $\boldsymbol{\mu}$ has a density then an *optimal* Monge map $T^\star$ between $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ must exist.

Prop. In that case

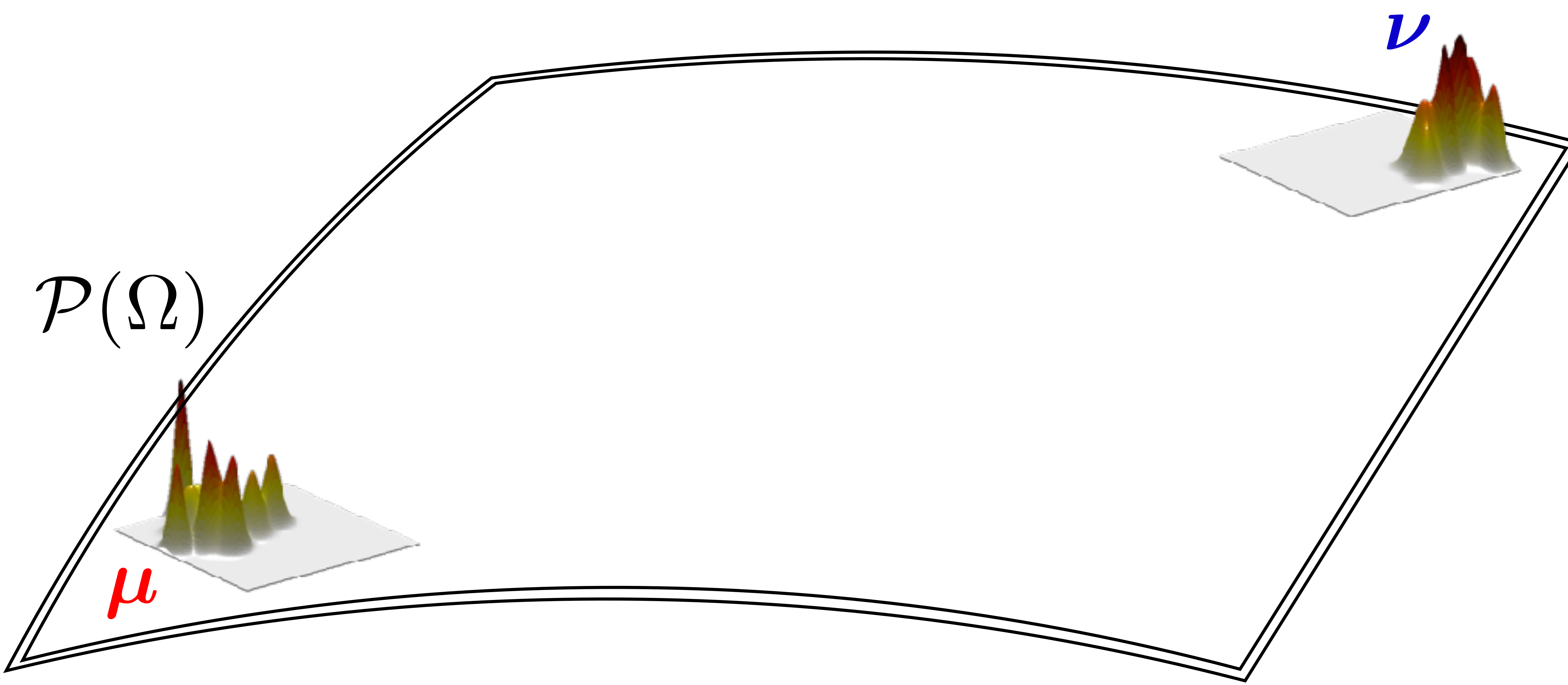
$$\boldsymbol{P}^\star := (\mathrm{Id}, T^\star)_\sharp \boldsymbol{\mu} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})$$

is also *optimal* for the Kantorovich problem.

[Brenier'91] [Smith&Knott'87] [McCann'01]

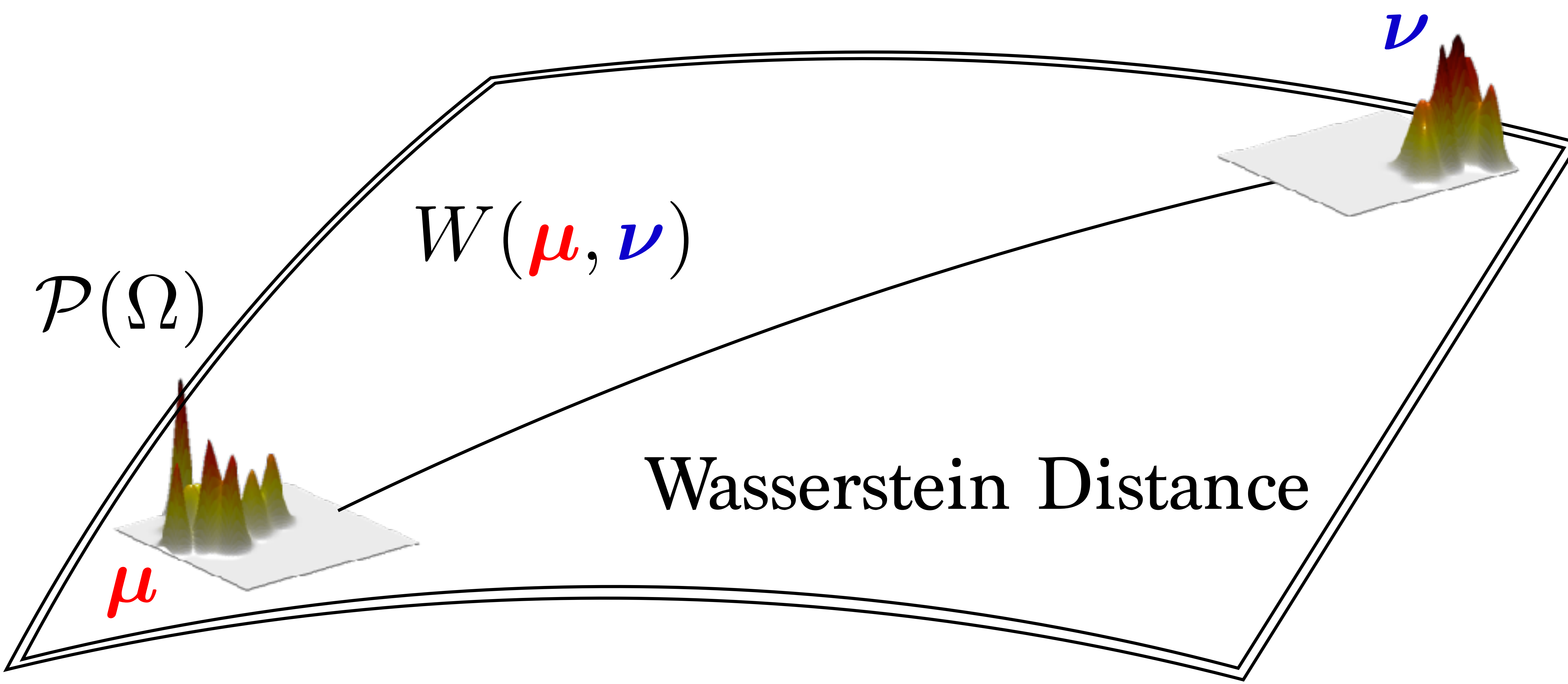
Optimal Transport Geometry

Very different geometry than standard information divergences (KL, Euclidean)



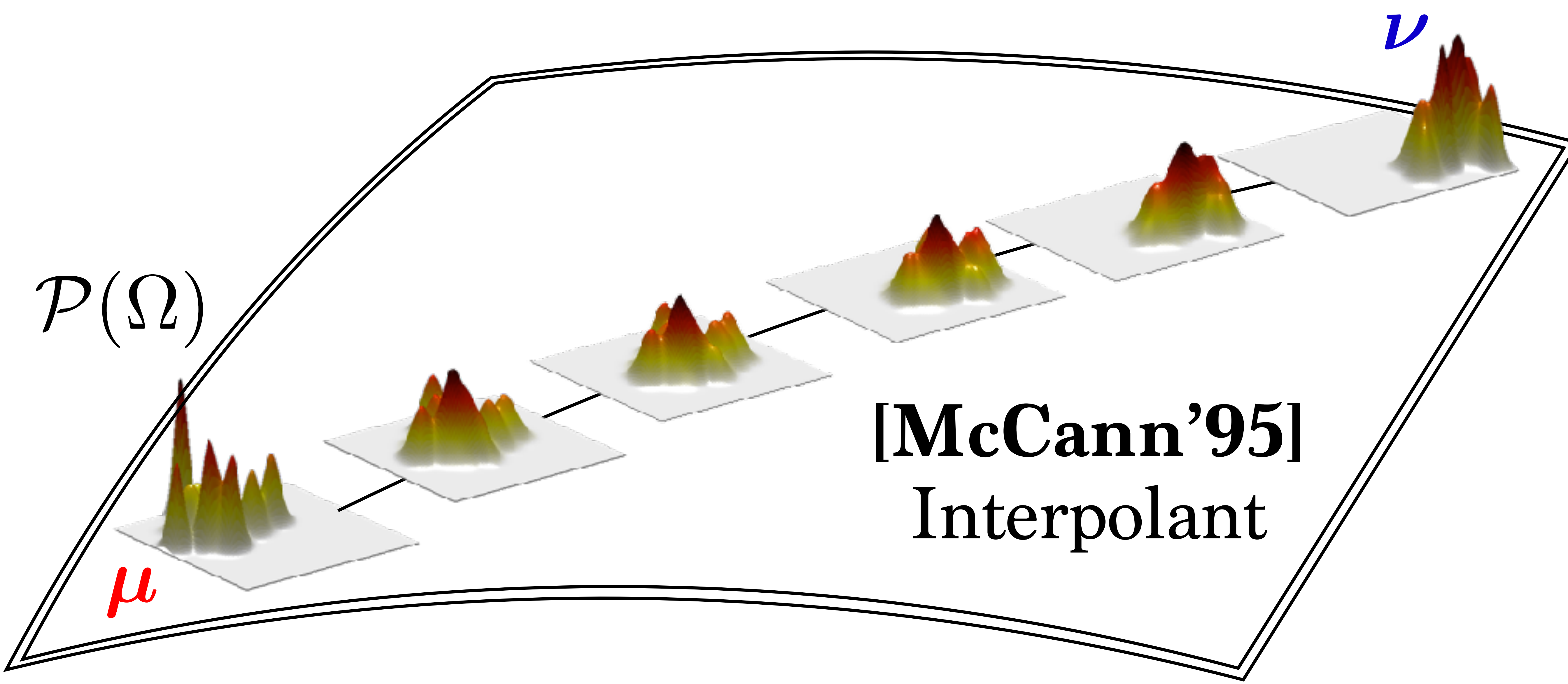
Optimal Transport Geometry

Very different geometry than standard information divergences (KL, Euclidean)



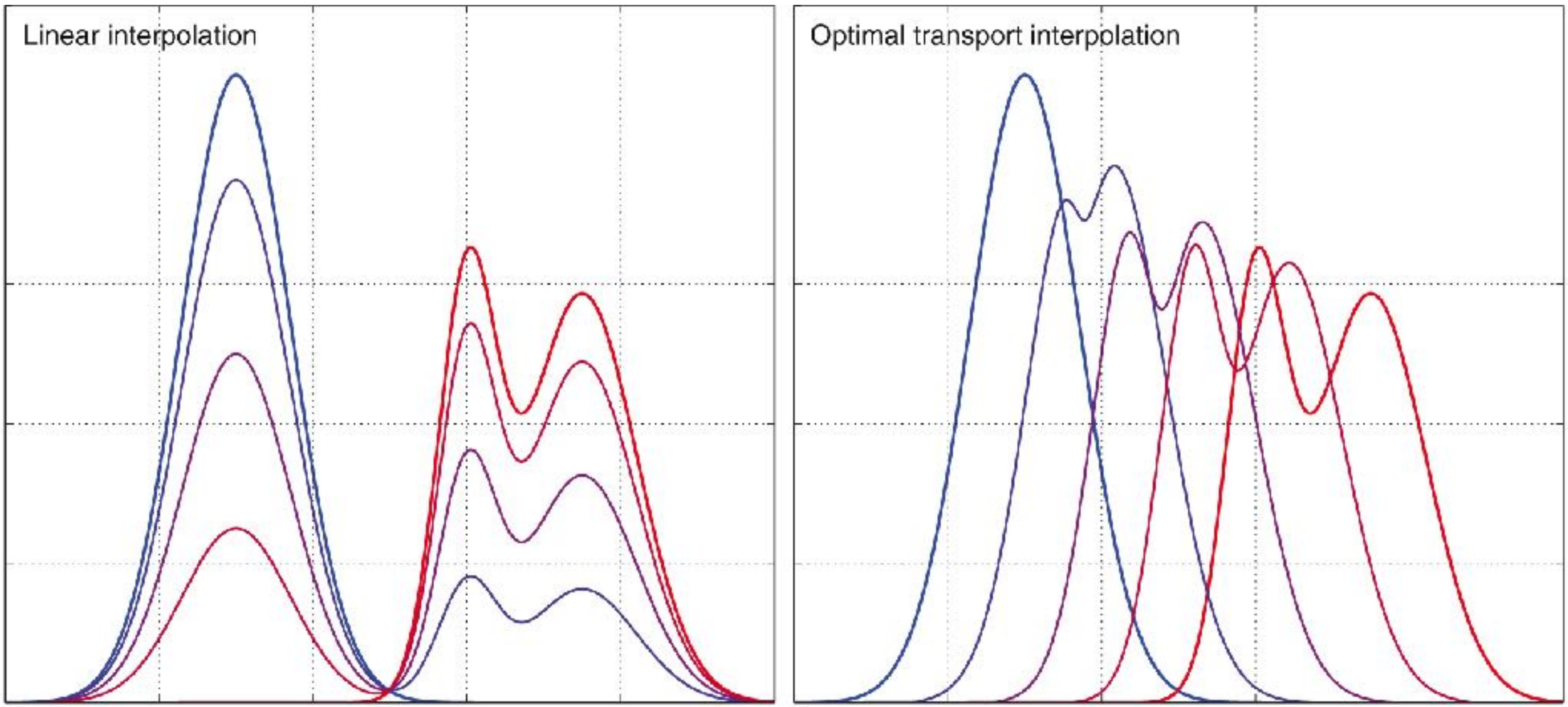
Optimal Transport Geometry

Very different geometry than standard information divergences (KL, Euclidean)



Optimal Transport Geometry

Very different geometry than standard information divergences (KL, Euclidean)



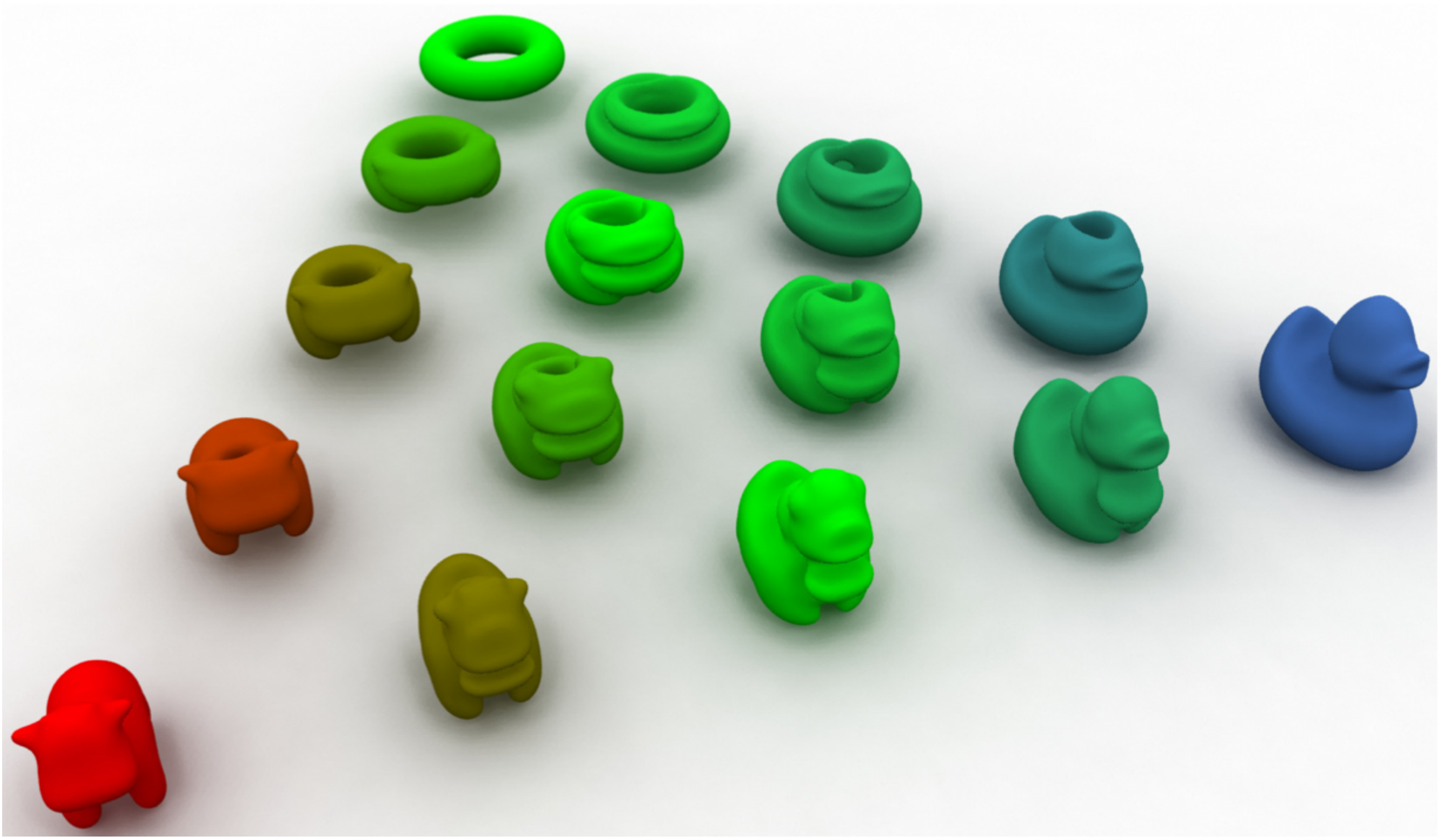
Optimal Transport Geometry

Very different geometry than standard information divergences (KL, Euclidean)

[SDPC..'15]

Optimal Transport Geometry

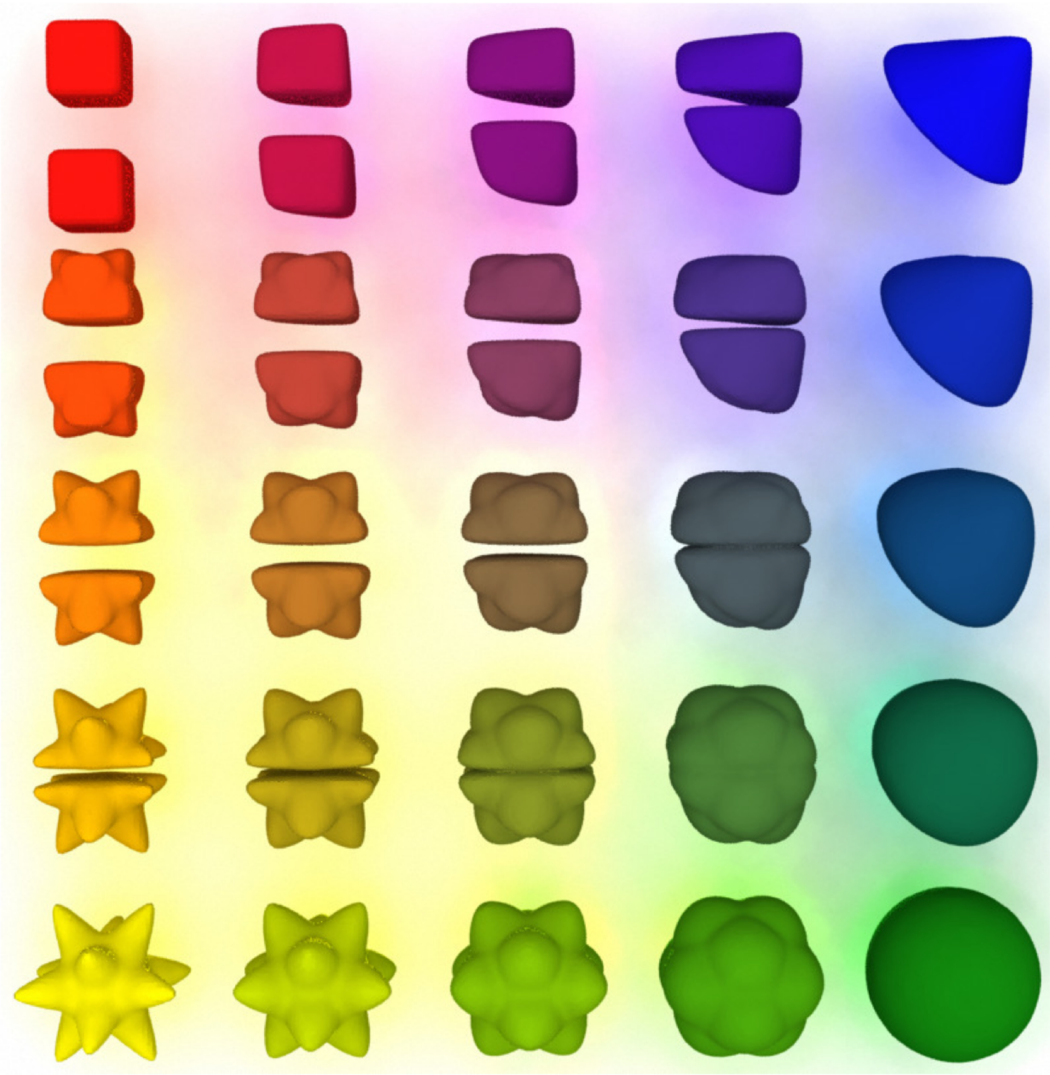
Very different geometry than standard information divergences (*KL*, Euclidean)



[SDPC..'15]

Optimal Transport Geometry

Very different geometry than standard information divergences (KL, Euclidean)



[SDPC..'15]

Variational OT Problems in ML

Up to 2010: OT solvers used mostly for retrieval in databases of histograms

$$W_p(\boldsymbol{\mu}, \boldsymbol{\nu}) = ?$$

$$W_p(\boldsymbol{\mu}, \boldsymbol{\nu}) \leq \cdots ?$$

OT is now used as a **loss** or **fidelity** term:

$$\operatorname*{argmin}_{\boldsymbol{\mu} \in \mathcal{P}(\Omega)} F(W_p(\boldsymbol{\mu}, \boldsymbol{\nu}_{\mathbf{1}}), W_p(\boldsymbol{\mu}, \boldsymbol{\nu}_{\mathbf{2}}), \dots, \boldsymbol{\mu}) = ?$$

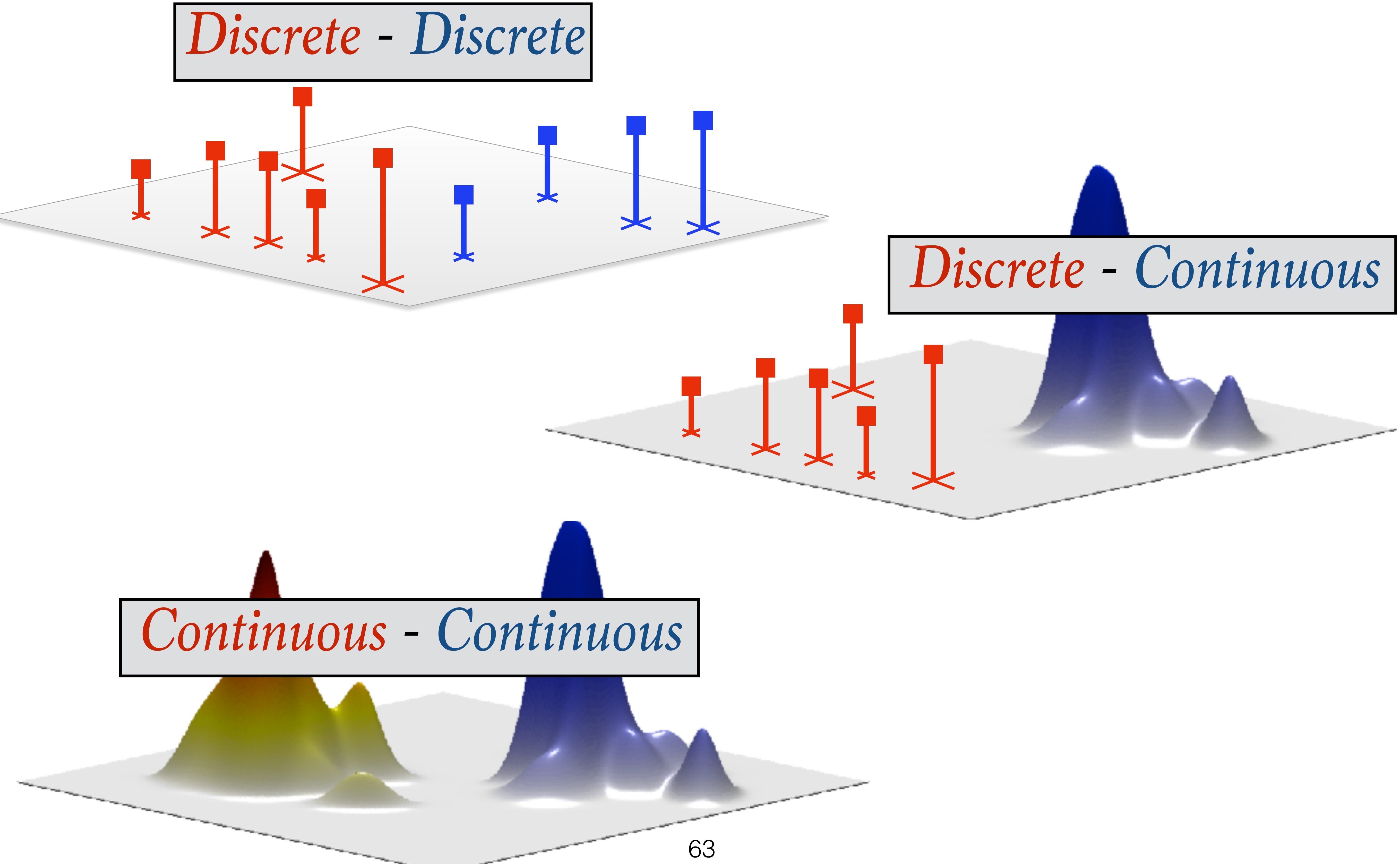
[Jordan Kinderlehrer Otto'98]
[Ambrosio Gigli Savaré'05]

$$\text{``}\nabla_{\boldsymbol{\mu}}\text{''} W_p(\boldsymbol{\mu}, \boldsymbol{\nu}) = ?$$

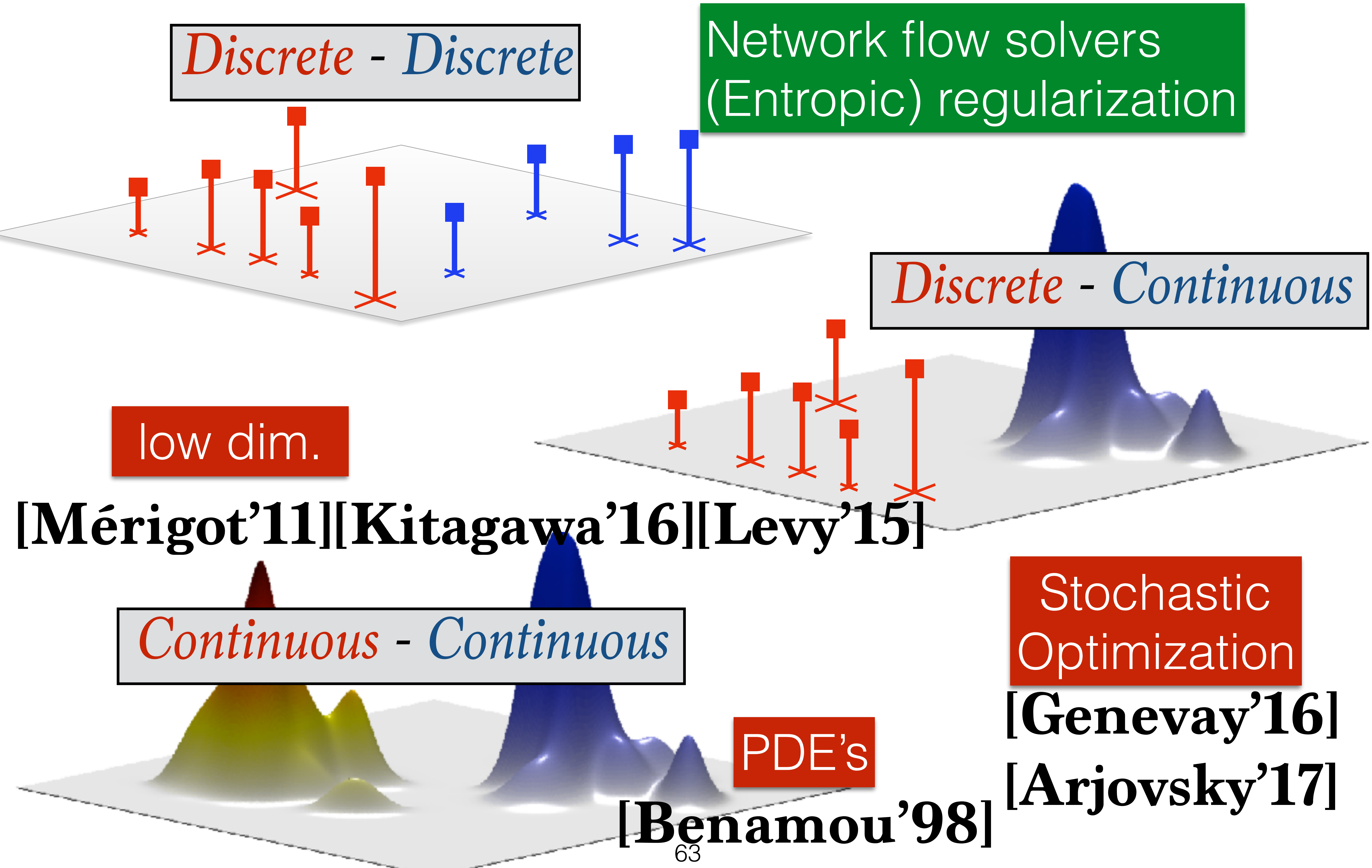
2. Computing OT exactly

- Typology: discrete/continuous problems
- Easy cases and exact solvers for discrete measures.

When do we compute OT?



When do we compute OT?



Easy (1): Univariate Measures

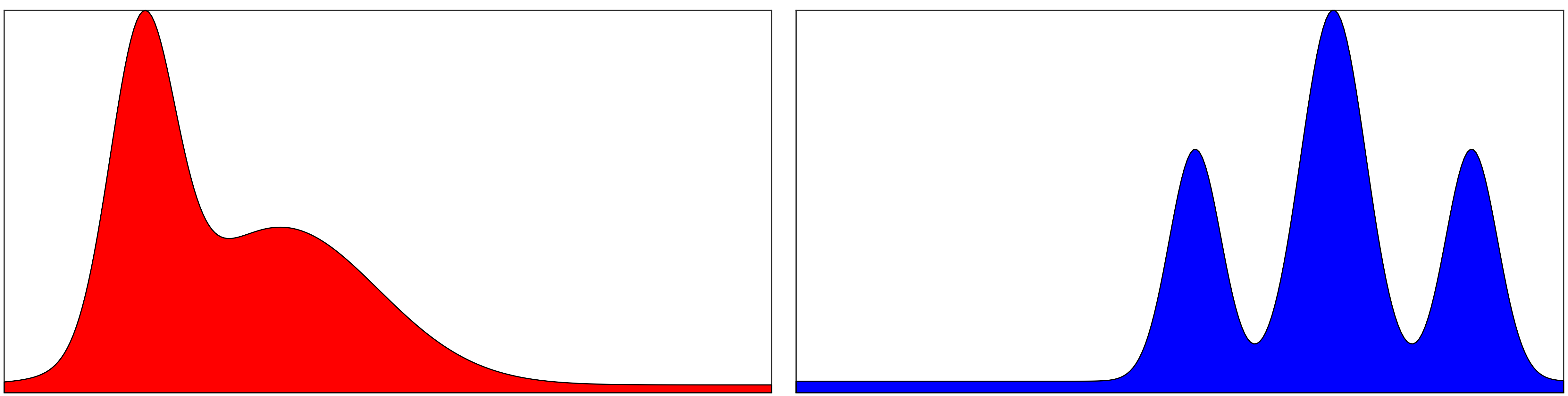
Remark. If $\Omega = \mathbb{R}$, $\boldsymbol{c}(x, y) = \boldsymbol{c}(|x - y|)$, $\boldsymbol{c}$ convex, $F_{\boldsymbol{\mu}}^{-1}, F_{\boldsymbol{\nu}}^{-1}$ quantile functions,

$$W(\boldsymbol{\mu}, \boldsymbol{\nu}) = \int_0^1 \boldsymbol{c}(|F_{\boldsymbol{\mu}}^{-1}(x) - F_{\boldsymbol{\nu}}^{-1}(x)|)dx$$

Easy (1): Univariate Measures

Remark. If $\Omega = \mathbb{R}$, $c(x, y) = c(|x - y|)$, c convex, $F_{\mu}^{-1}, F_{\nu}^{-1}$ quantile functions,

$$W(\mu, \nu) = \int_0^1 c(|F_{\mu}^{-1}(x) - F_{\nu}^{-1}(x)|)dx$$

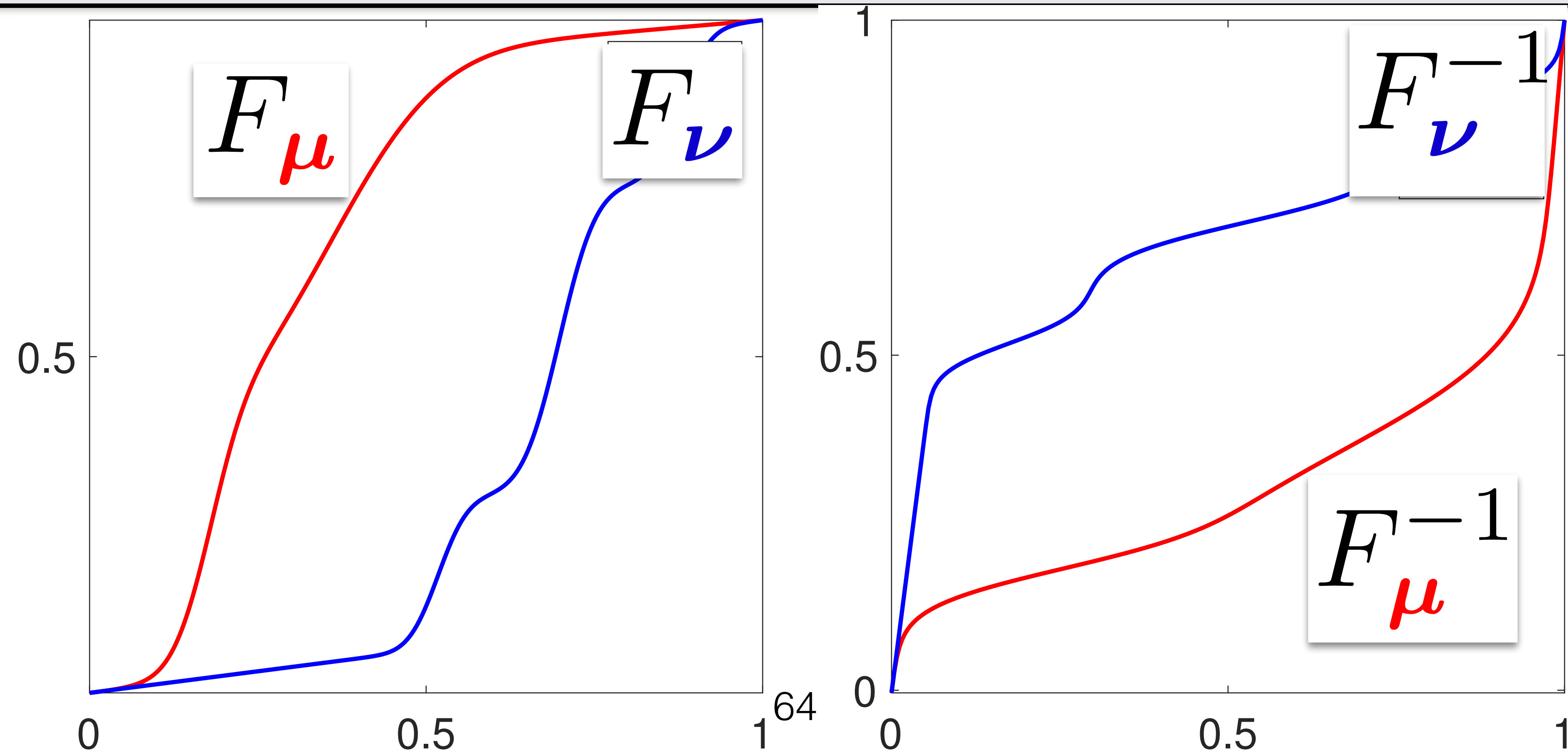


μ ν

Easy (1): Univariate Measures

Remark. If $\Omega = \mathbb{R}$, $c(x, y) = c(|x - y|)$, c convex, F_μ^{-1}, F_ν^{-1} quantile functions,

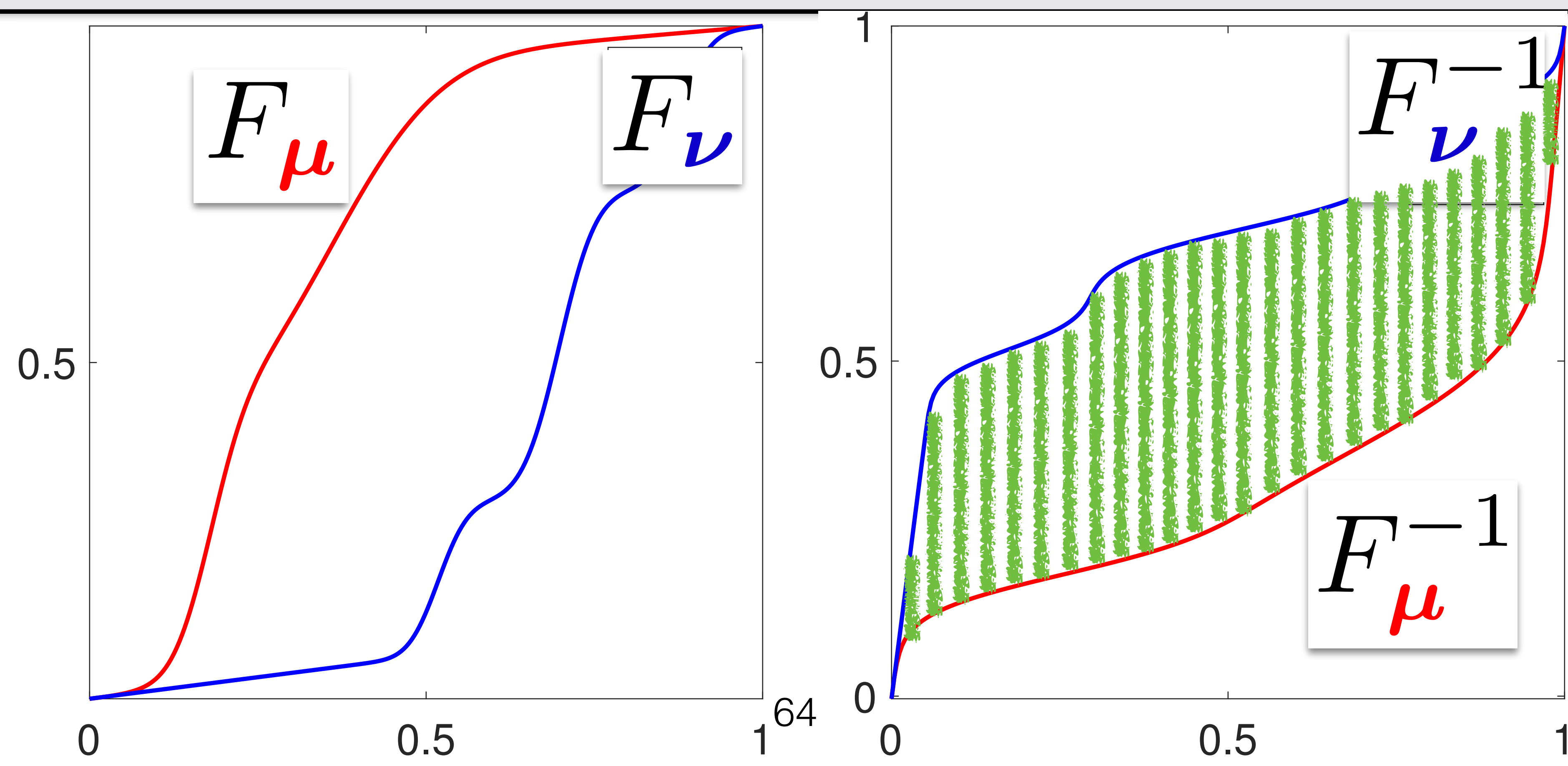
$$W(\mu, \nu) = \int_0^1 c(|F_\mu^{-1}(x) - F_\nu^{-1}(x)|)dx$$



Easy (1): Univariate Measures

Remark. If $\Omega = \mathbb{R}$, $c(x,y) = c(|x-y|)$, c convex, $F_{\mu}^{-1}, F_{\nu}^{-1}$ quantile functions,

$$W(\mu,\nu) = \int_0^1 c(|F_{\mu}^{-1}(x) - F_{\nu}^{-1}(x)|)dx$$



Easy (2): Gaussian Measures

Remark. If $\Omega = \mathbb{R}^d$, $\boldsymbol{c}(x, y) = \|x - y\|^2$, and $\boldsymbol{\mu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\mu}}), \boldsymbol{\nu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\nu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})$ then

$$W_2^2(\boldsymbol{\mu}, \boldsymbol{\nu}) = \|\mathbf{m}_{\boldsymbol{\mu}} - \mathbf{m}_{\boldsymbol{\nu}}\|^2 + B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2$$

where B is the Bures metric

$$B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2 = \mathrm{trace}(\boldsymbol{\Sigma}_{\boldsymbol{\mu}} + \boldsymbol{\Sigma}_{\boldsymbol{\nu}} - 2(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2})^{1/2}).$$

Easy (2): Gaussian Measures

Remark. If $\Omega = \mathbb{R}^d$, $\boldsymbol{c}(x,y) = \|x-y\|^2$, and $\boldsymbol{\mu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\mu}}), \boldsymbol{\nu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\nu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})$ then

$$W_2^2(\boldsymbol{\mu}, \boldsymbol{\nu}) = \|\mathbf{m}_{\boldsymbol{\mu}} - \mathbf{m}_{\boldsymbol{\nu}}\|^2 + B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2$$

where B is the Bures metric

$$B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2 = \mathrm{trace}(\boldsymbol{\Sigma}_{\boldsymbol{\mu}} + \boldsymbol{\Sigma}_{\boldsymbol{\nu}} - 2(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2})^{1/2}).$$

The map $T : x \mapsto \mathbf{m}_{\boldsymbol{\nu}} + A(x - \mathbf{m}_{\boldsymbol{\mu}})$ is **optimal**, where $A = \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-\frac{1}{2}} \left(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{\frac{1}{2}} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{\frac{1}{2}} \right)^{\frac{1}{2}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-\frac{1}{2}}$.

Easy (2): Gaussian Measures

Remark. If $\Omega = \mathbb{R}^d$, $c(x,y) = \|x-y\|^2$, and $\mu = \mathcal{N}(\mathbf{m}_{\mu}, \mathbf{\Sigma}_{\mu})$, $\nu = \mathcal{N}(\mathbf{m}_{\nu}, \mathbf{\Sigma}_{\nu})$ then

$$W_2^2(\mu,\nu) = \|\mathbf{m}_{\mu} - \mathbf{m}_{\nu}\|^2 + B(\mathbf{\Sigma}_{\mu}, \mathbf{\Sigma}_{\nu})^2$$

where B is the Bures

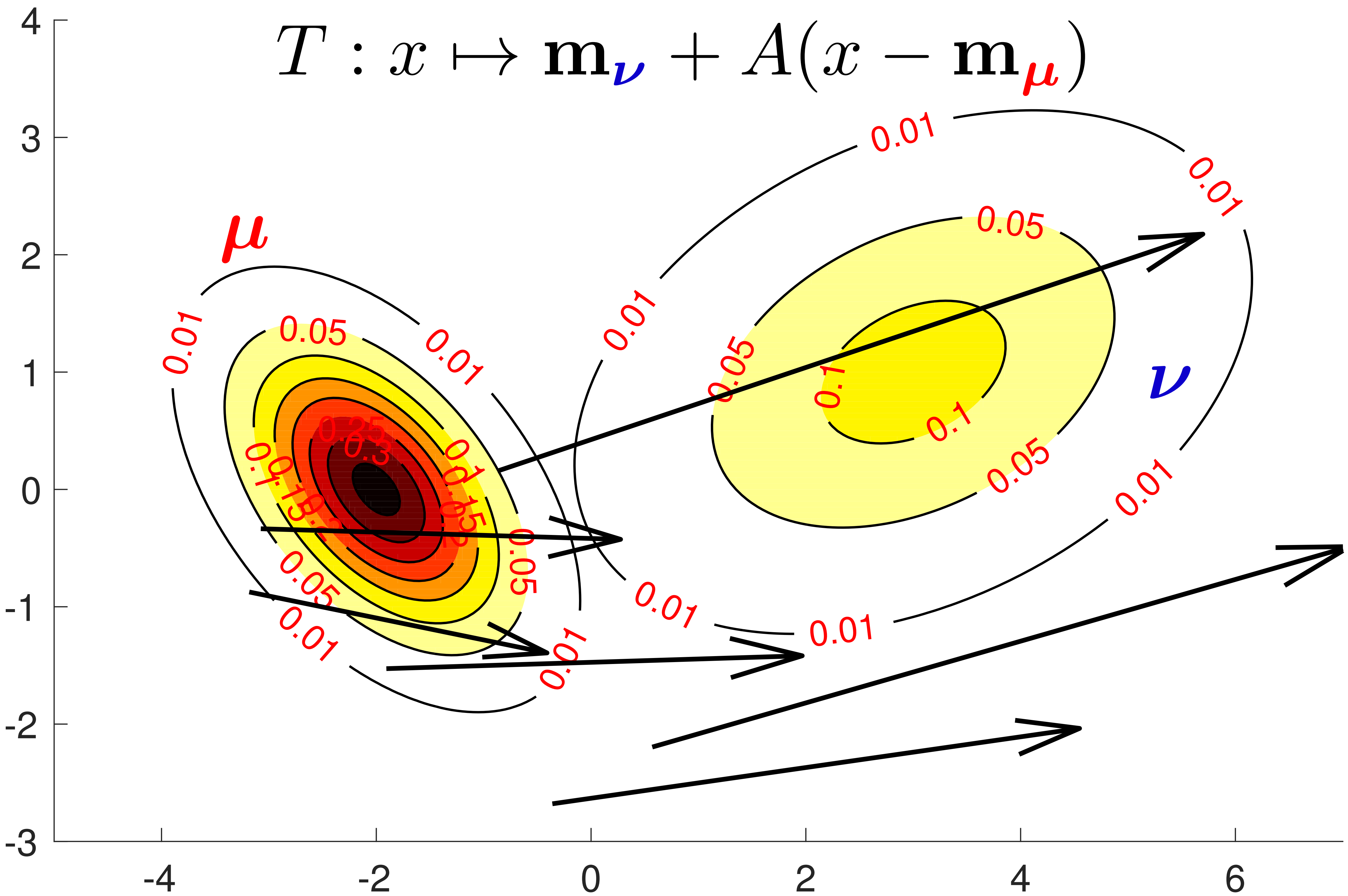
if $X \sim \mathcal{N}(m, \mathbf{\Sigma})$ then $Y := CX + b \sim \mathcal{N}(Cm + b, C\mathbf{\Sigma}C^T)$.

$\operatorname{trace}(\mathbf{\Sigma}_{\mu}+\mathbf{\Sigma}_{\nu}-2(\mathbf{\Sigma}_{\mu}^{1/2}\mathbf{\Sigma}_{\nu}\mathbf{\Sigma}_{\mu}^{1/2})^{1/2})$.

The map $T : x \mapsto \mathbf{m}_{\nu} + A(x - \mathbf{m}_{\mu})$ is **optimal**,

where $A = \mathbf{\Sigma}_{\mu}^{-\frac{1}{2}}\left(\mathbf{\Sigma}_{\mu}^{\frac{1}{2}}\mathbf{\Sigma}_{\nu}\mathbf{\Sigma}_{\mu}^{\frac{1}{2}}\right)^{\frac{1}{2}}\mathbf{\Sigma}_{\mu}^{-\frac{1}{2}}$.

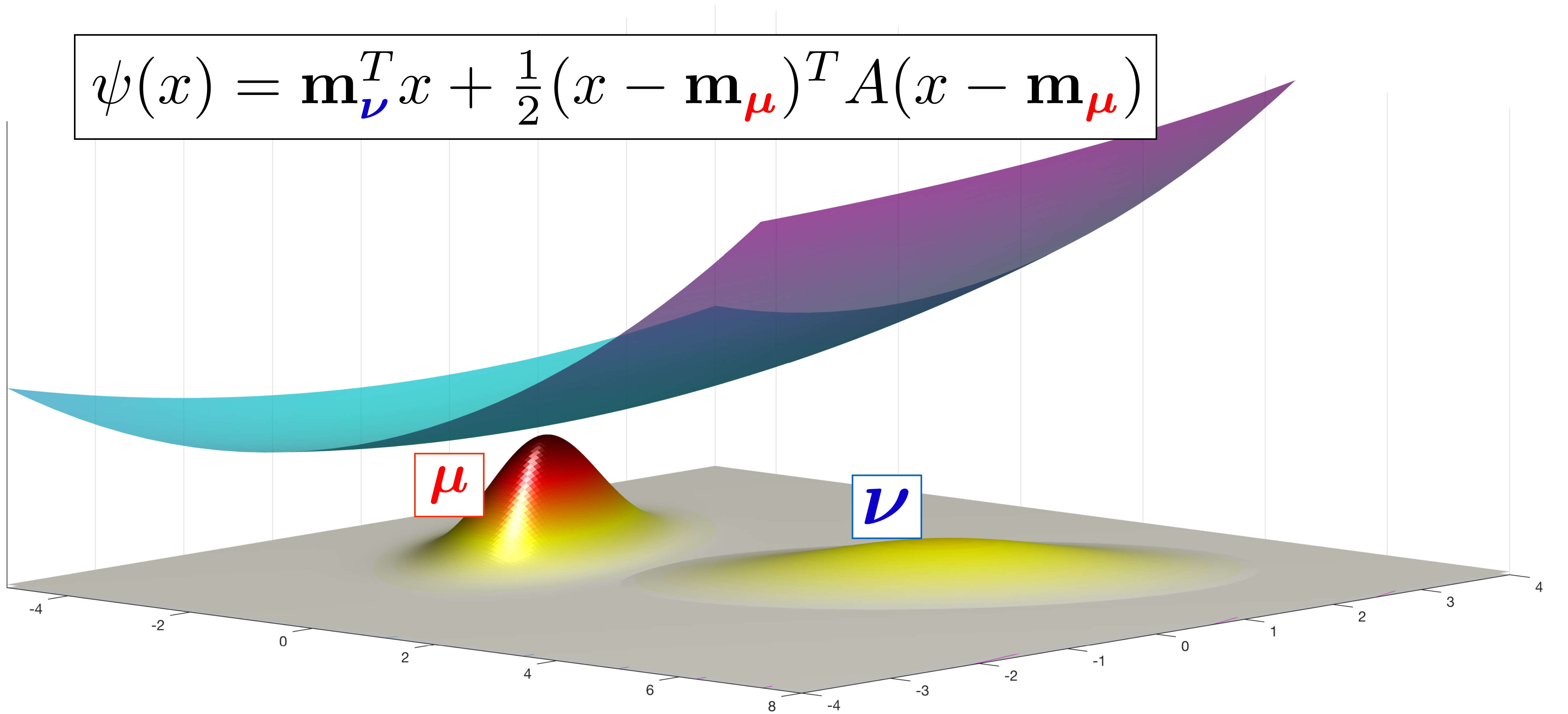
Easy (2): Gaussian Measures



Easy (2): Gaussian Measures

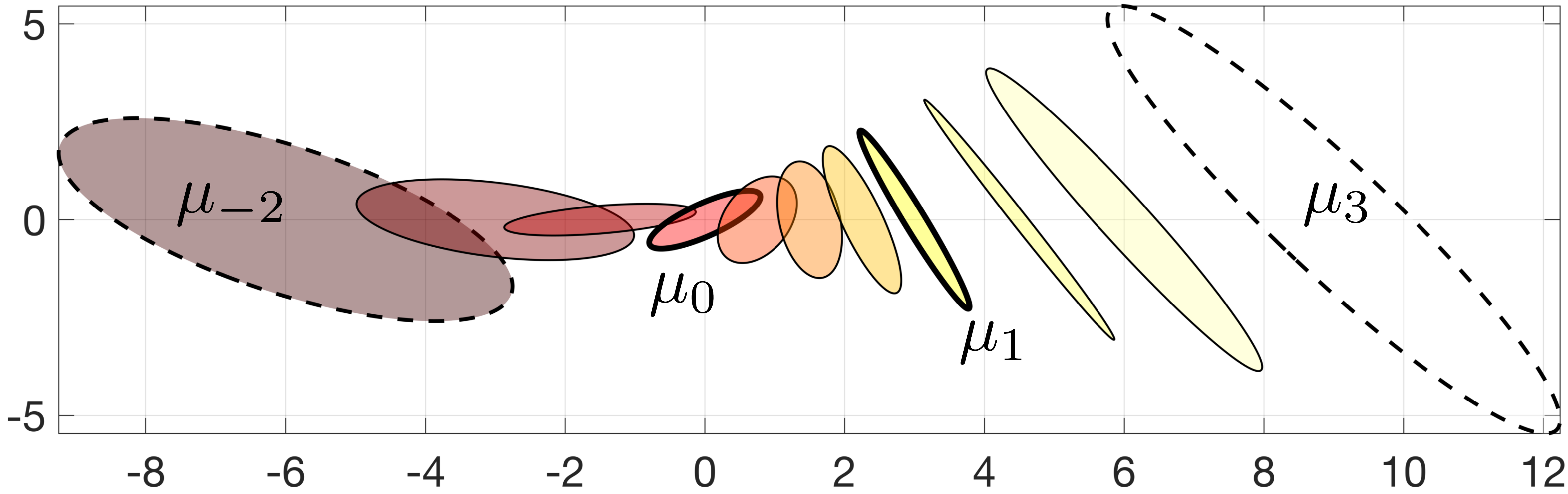
$$T = \nabla\psi : x \mapsto \mathbf{m}_{\boldsymbol{\nu}} + A(x - \mathbf{m}_{\boldsymbol{\mu}})$$

$$\boxed{\psi(x) = \mathbf{m}_{\boldsymbol{\nu}}^T x + \frac{1}{2}(x - \mathbf{m}_{\boldsymbol{\mu}})^T A (x - \mathbf{m}_{\boldsymbol{\mu}})}$$



Easy (2): Gaussian Measures

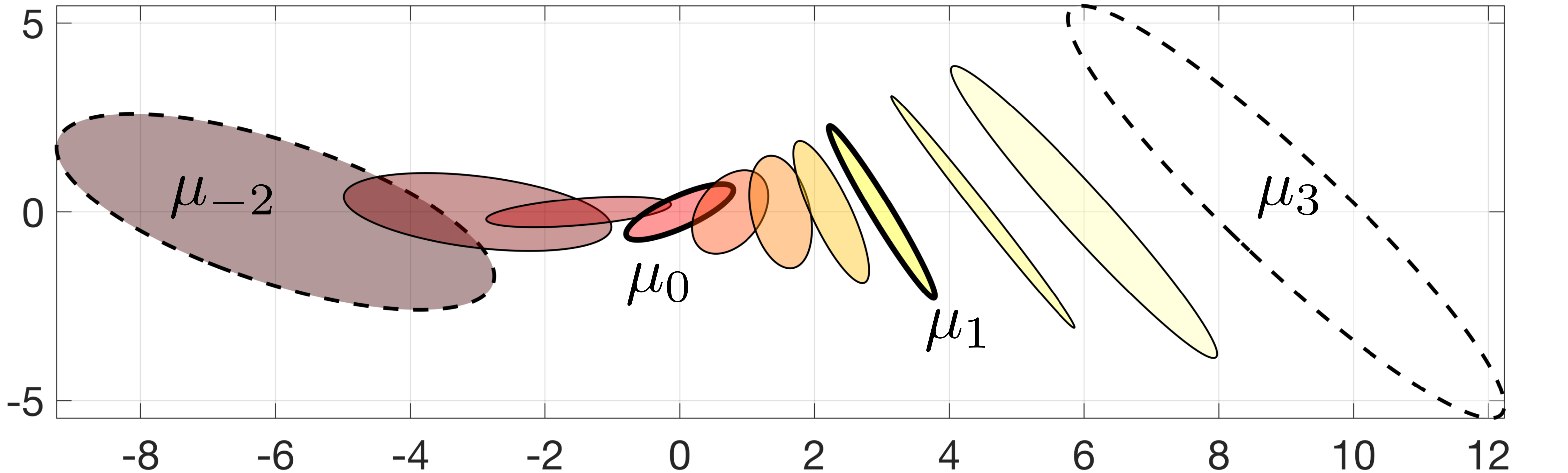
W_2 geodesic $(\mu_t)_t$ from μ_0 to μ_1 ($t \in [0,1]$) and extrapolation



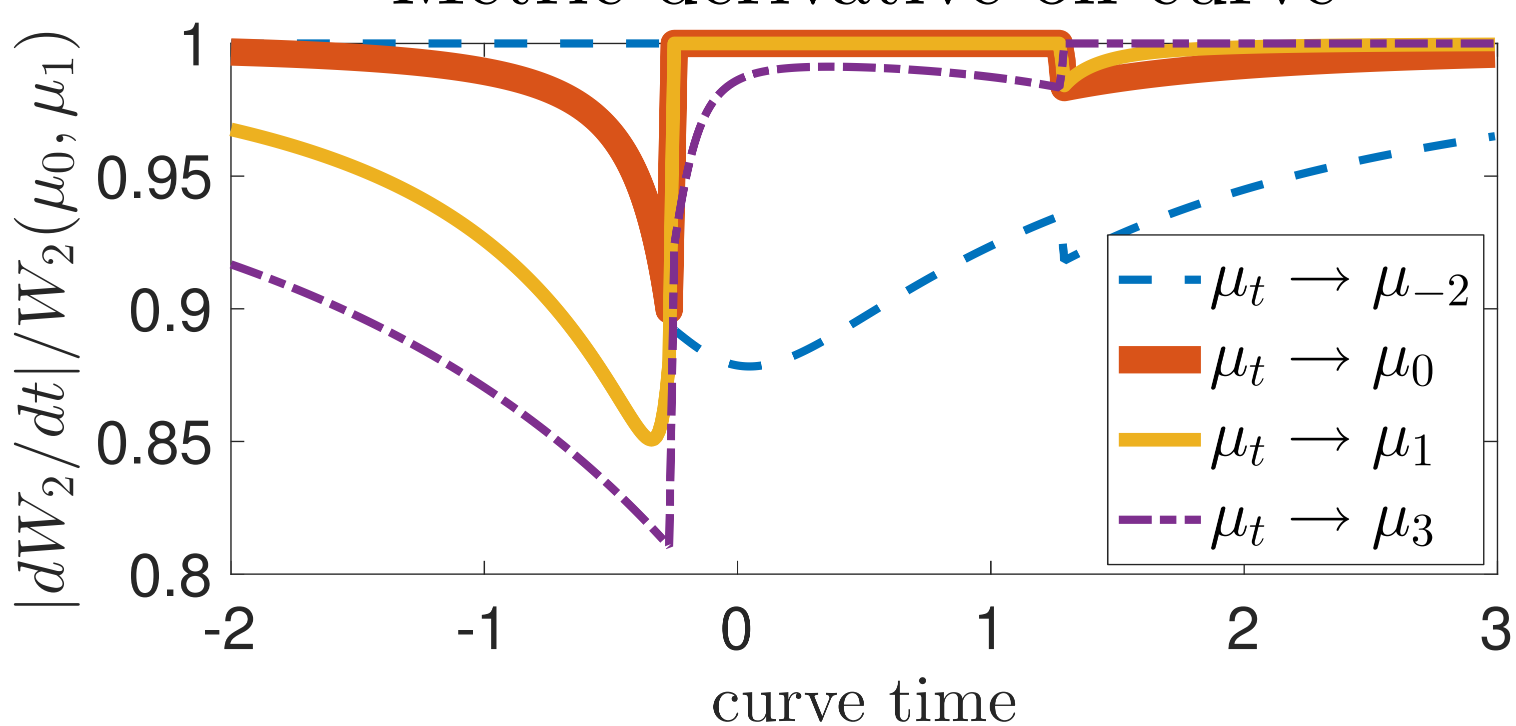
$$\Sigma_t = ((1-t)\,I + tA)\,\Sigma_\mu\,((1-t)\,I + tA)$$

Easy (2): Gaussian Measures

W_2 geodesic $(\mu_t)_t$ from μ_0 to μ_1 ($t \in [0,1]$) and extrapolation



Metric derivative on curve



Easy (3): Elliptical Distributions

$$T = \nabla\psi : x \mapsto \mathbf{m}_{\boldsymbol{\nu}} + A(x - \mathbf{m}_{\boldsymbol{\mu}})$$

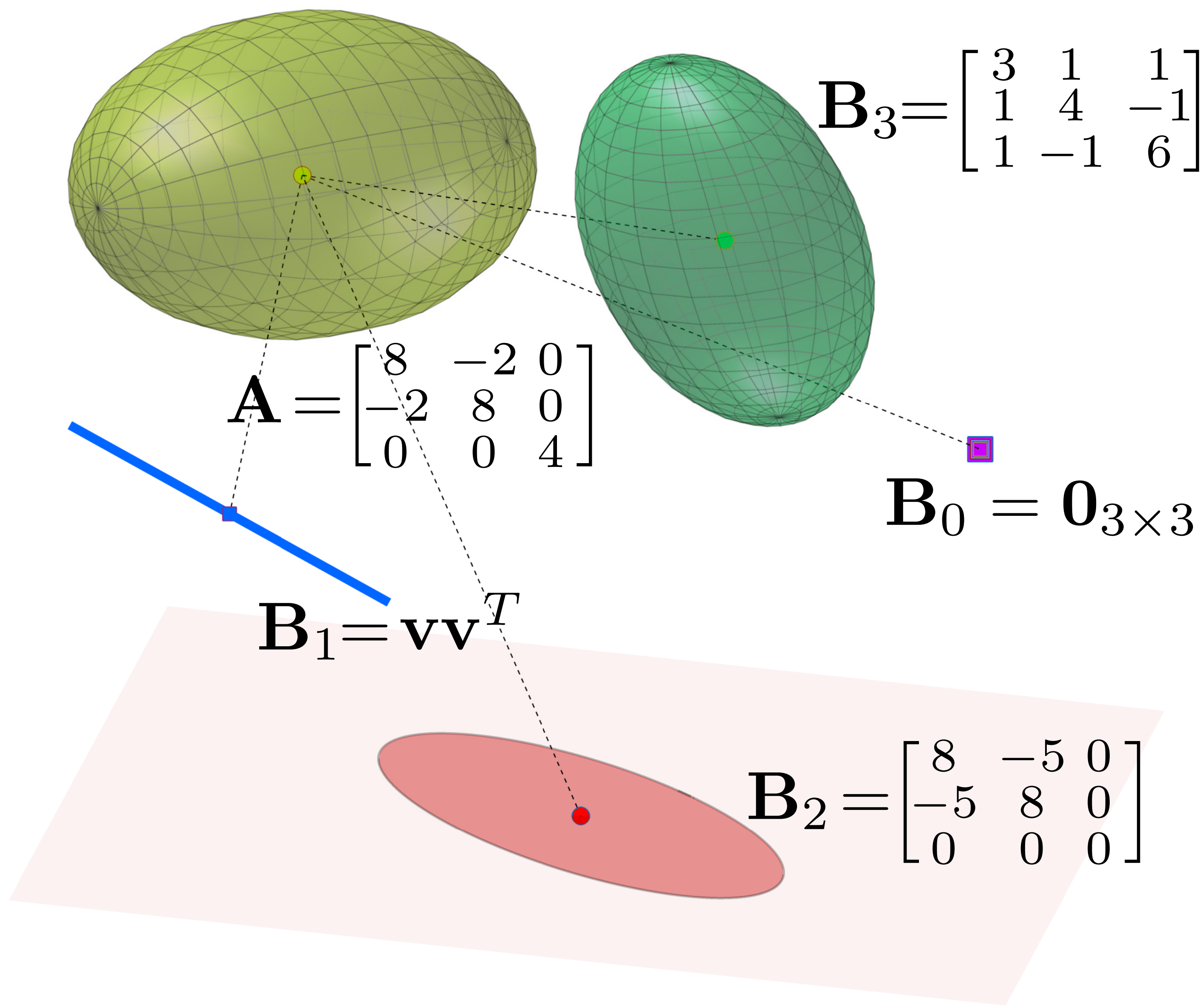
[Gelbrich'92] shows that the linear map T is also **optimal** for elliptically contoured distributions, *i.e.* distributions whose MGF are

$$\phi_X(\mathbf{t}) = \mathbb{E}\left[e^{\sqrt{-1}\mathbf{t}^T X}\right] = e^{\sqrt{-1}\mathbf{t}^T \mathbf{m}} g(\mathbf{t}^T \boldsymbol{C}\mathbf{t})$$

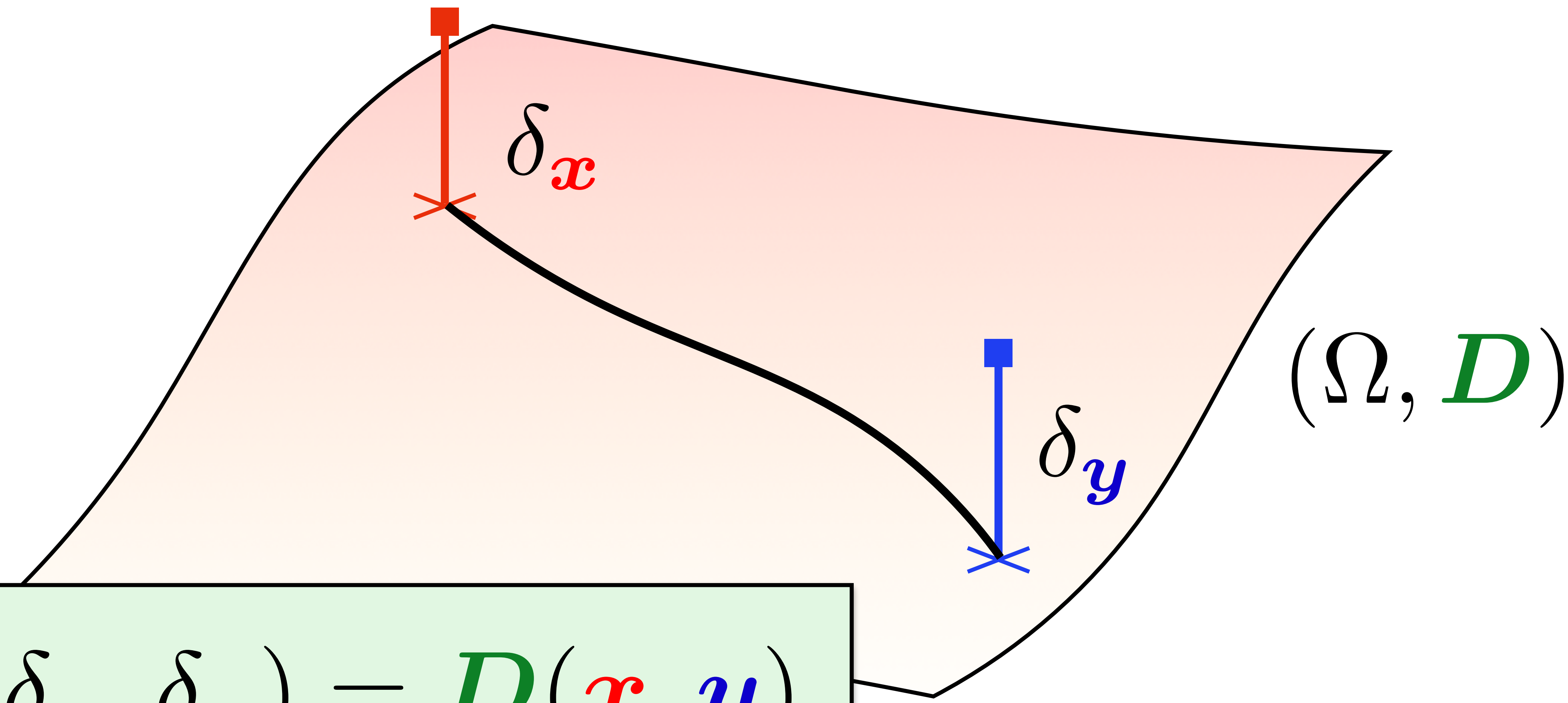
g of positive type.

Same formula applies, but variance is a factor (depends on g) of **C**, hence Bures factor is scaled.

Easy (3): Uniform Ellipses

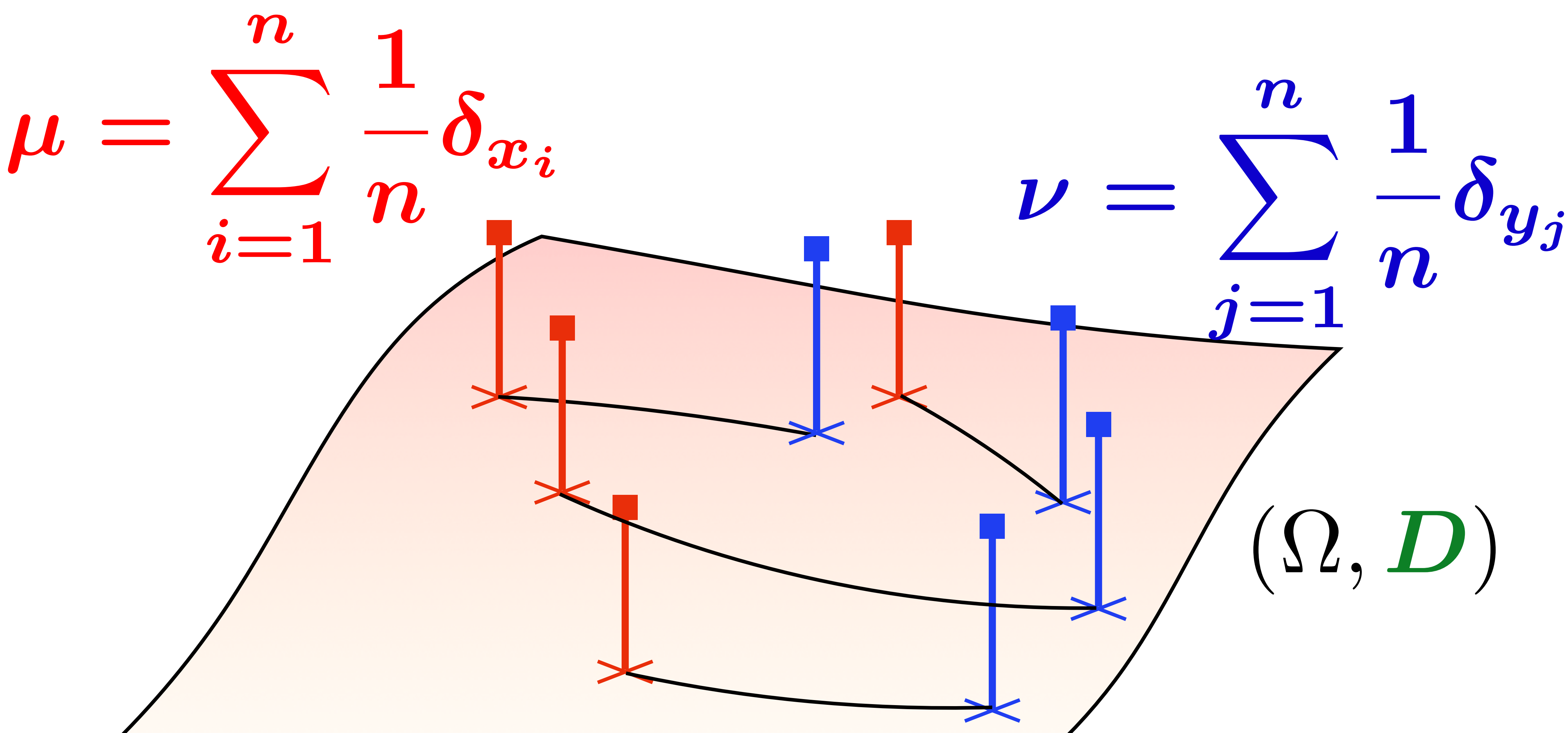


Wasserstein Between Two Diracs



$$W_p(\delta_{\boldsymbol{x}}, \delta_{\boldsymbol{y}}) = \boldsymbol{D}(\boldsymbol{x}, \boldsymbol{y})$$

Linear Assignment ⊂ Wasserstein

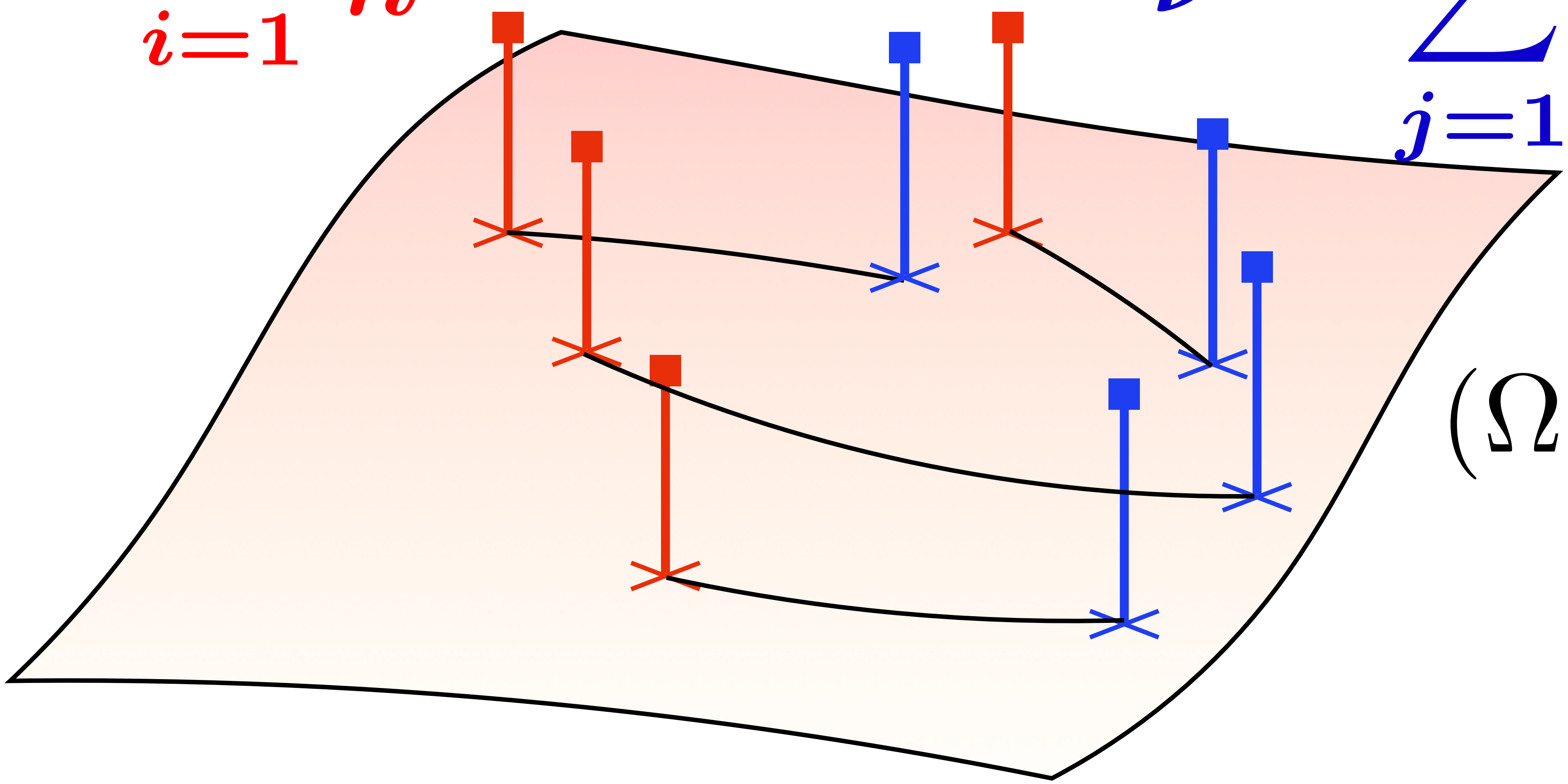


$$W_p^p(\mu, \nu) = \min_{\sigma \in S_n} \frac{1}{n} \sum_{i=1}^{n} D(x_i, y_{\sigma_i})^p$$

Linear Assignment $\subset$ Wasserstein

$$\mu = \sum_{i=1}^{n} \frac{1}{n}\delta_{x_i}$$

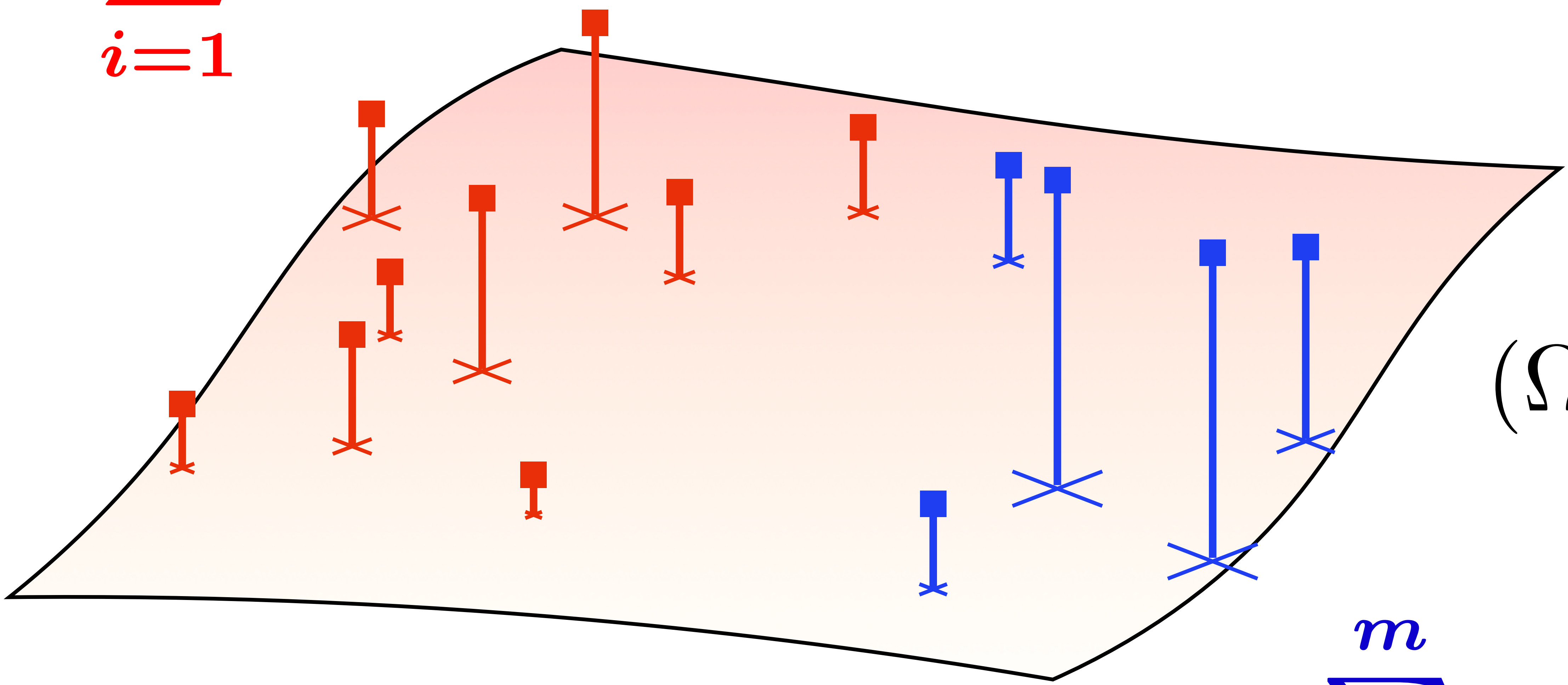
$$\nu = \sum_{j=1}^{n} \frac{1}{n}\delta_{y_j}$$



(Ω, D)

OT on Two Empirical Measures

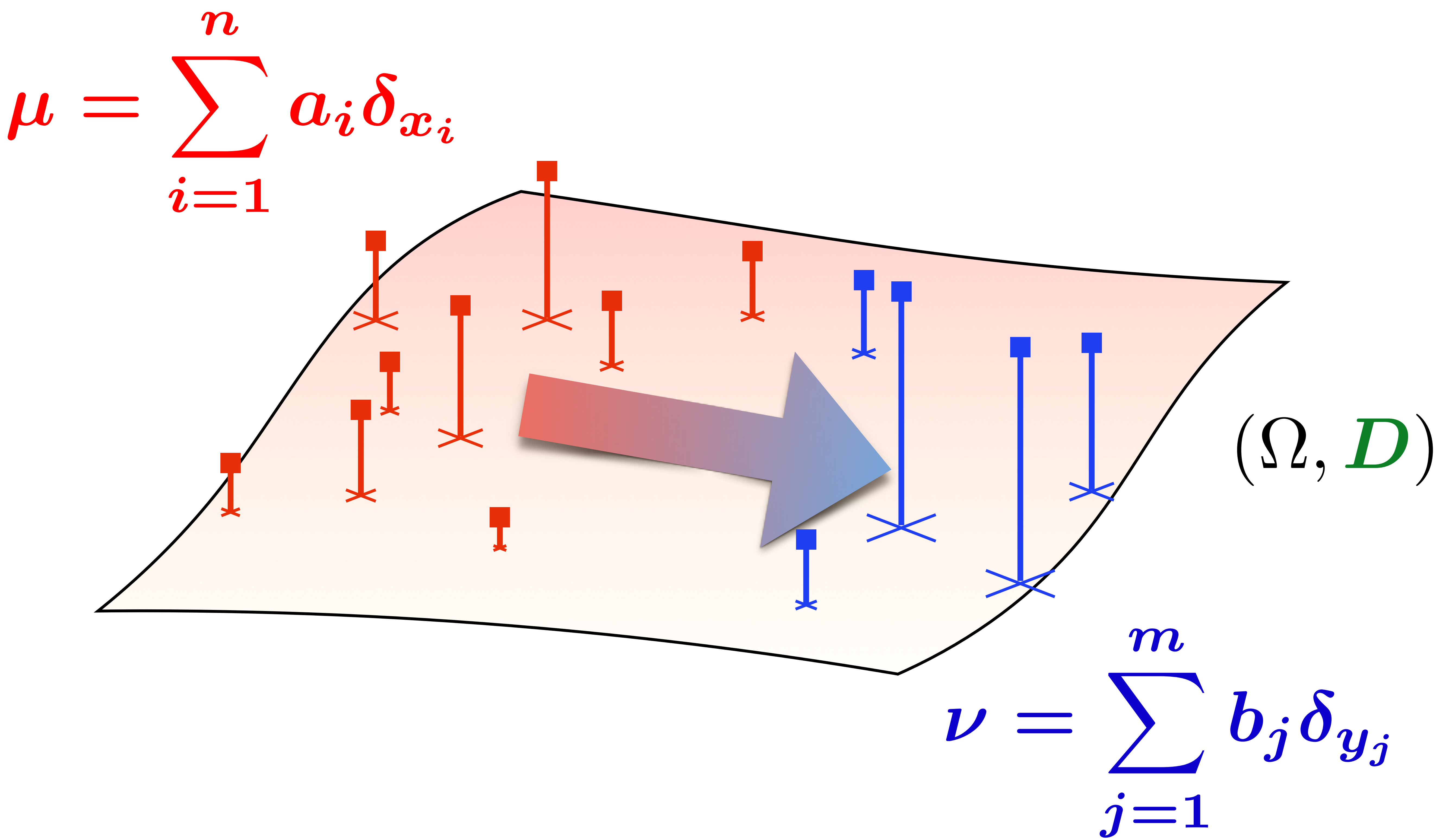
$$\mu = \sum_{i=1}^{n} a_i \delta_{x_i}$$



(Ω, D)

$$\nu = \sum_{j=1}^{m} b_j \delta_{y_j}$$

OT on Two Empirical Measures



Wasserstein on Empirical Measures

Consider $\boldsymbol{\mu} = \sum_{i=1}^{n} \boldsymbol{a_i \delta_{x_i}}$ and $\boldsymbol{\nu} = \sum_{j=1}^{m} \boldsymbol{b_j \delta_{y_j}}$.

$$M_{\mathbf{XY}} \stackrel{\text{def}}{=} [D(\boldsymbol{x}_i, \boldsymbol{y}_j)^p]_{ij}$$

$$U(\boldsymbol{a}, \boldsymbol{b}) \stackrel{\text{def}}{=} \{\boldsymbol{P} \in \mathbb{R}_+^{n \times m} | \boldsymbol{P}\mathbf{1}_m = \boldsymbol{a}, \boldsymbol{P}^T\mathbf{1}_n = \boldsymbol{b}\}$$

Def. Optimal Transport Problem

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \min_{\boldsymbol{P} \in U(\boldsymbol{a}, \boldsymbol{b})} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle$$

Dual Kantorovich Problem

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \min_{\substack{\boldsymbol{P} \in \mathbb{R}_+^{n \times m} \\ \boldsymbol{P}\mathbf{1}_m = \boldsymbol{a}, \boldsymbol{P}^T\mathbf{1}_n = \boldsymbol{b}}} \langle \boldsymbol{P}, M_{\boldsymbol{XY}} \rangle$$

Dual Kantorovich Problem

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \min_{\substack{\boldsymbol{P} \in \mathbb{R}_+^{n \times m} \\ \boldsymbol{P}\mathbf{1}_m = \boldsymbol{a}, \boldsymbol{P}^T\mathbf{1}_n = \boldsymbol{b}}} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle$$

Def. Dual OT problem

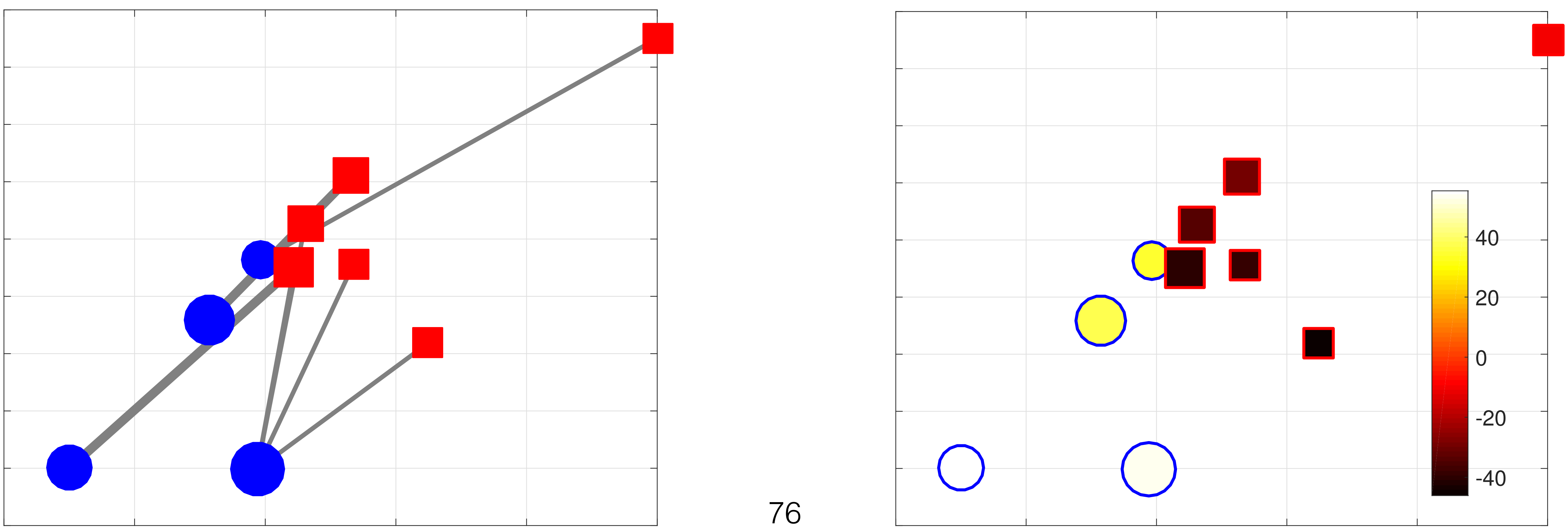
$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \max_{\substack{\boldsymbol{\alpha} \in \mathbb{R}^n, \boldsymbol{\beta} \in \mathbb{R}^m \\ \boldsymbol{\alpha}_{\boldsymbol{i}} + \boldsymbol{\beta}_{\boldsymbol{j}} \leq D(\boldsymbol{x}_{\boldsymbol{i}}, \boldsymbol{y}_{\boldsymbol{j}})^p}} \alpha^T \boldsymbol{a} + \beta^T \boldsymbol{b}$$

Dual Kantorovich Problem

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \min_{\substack{\boldsymbol{P}\in\mathbb{R}_+^{n\times m}\\ \boldsymbol{P}\mathbf{1}_m=\boldsymbol{a},\boldsymbol{P}^T\mathbf{1}_n=\boldsymbol{b}}} \langle \boldsymbol{P}, M_{\boldsymbol{XY}}\rangle$$

Def. Dual OT problem

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\substack{\boldsymbol{\alpha}\in\mathbb{R}^n,\boldsymbol{\beta}\in\mathbb{R}^m\\ \boldsymbol{\alpha}_{\boldsymbol{i}}+\boldsymbol{\beta}_{\boldsymbol{j}}\leq D(\boldsymbol{x}_{\boldsymbol{i}},\boldsymbol{y}_{\boldsymbol{j}})^p}} \alpha^T\boldsymbol{a}+\beta^T\boldsymbol{b}$$

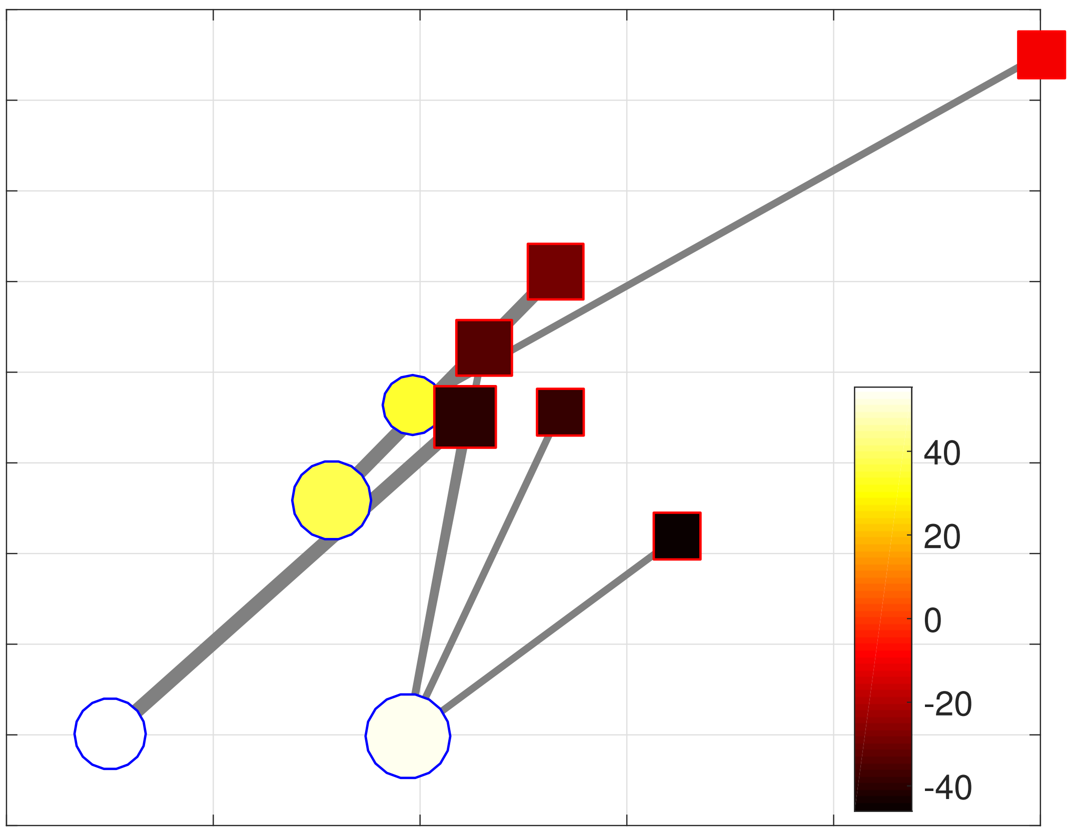


Dual Kantorovich Problem

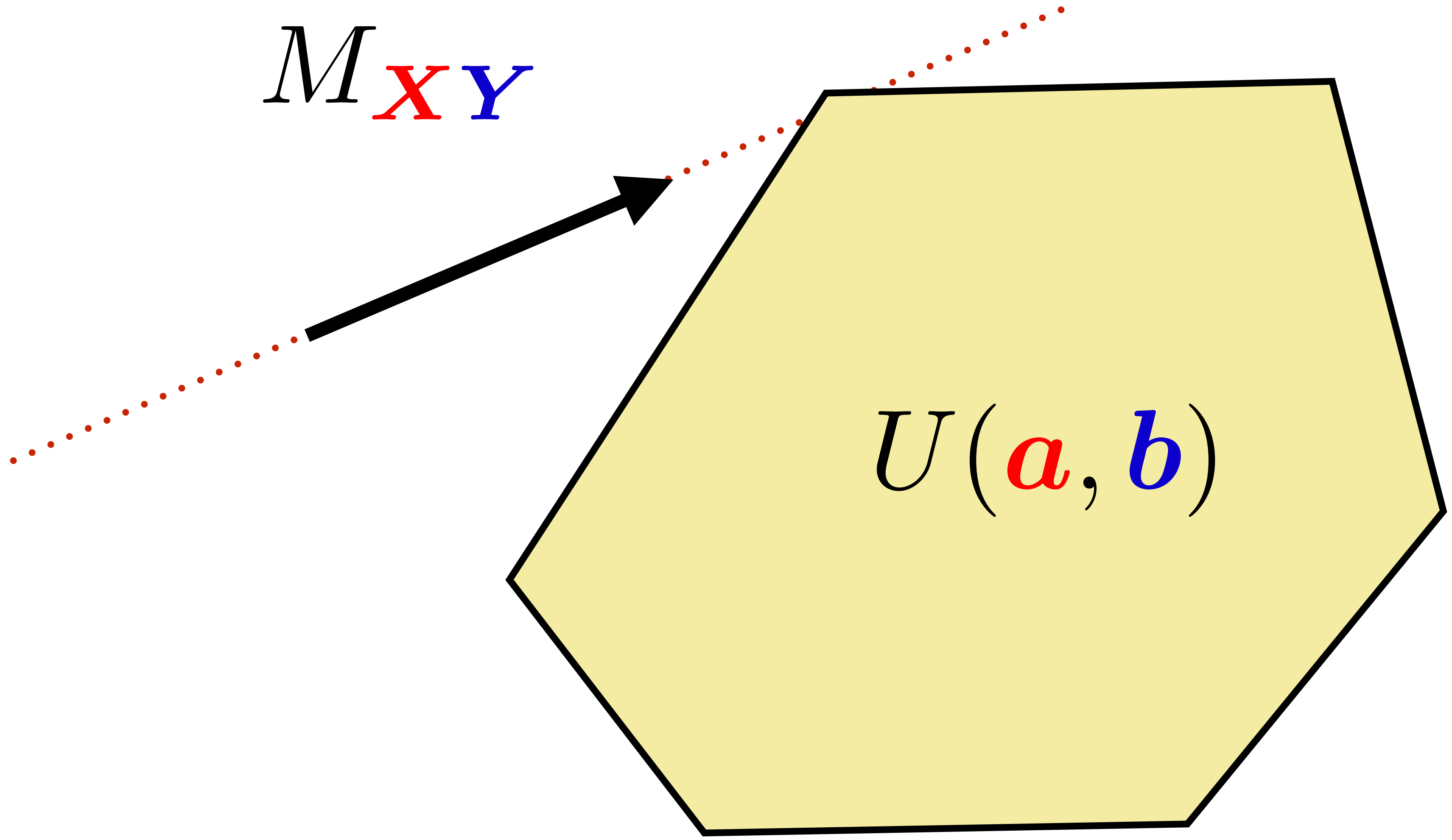
$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \min_{\substack{\boldsymbol{P} \in \mathbb{R}_+^{n \times m} \\ \boldsymbol{P}\mathbf{1}_m = \boldsymbol{a}, \boldsymbol{P}^T \mathbf{1}_n = \boldsymbol{b}}} \langle \boldsymbol{P}, M_{\boldsymbol{XY}} \rangle$$

Def. Dual OT problem

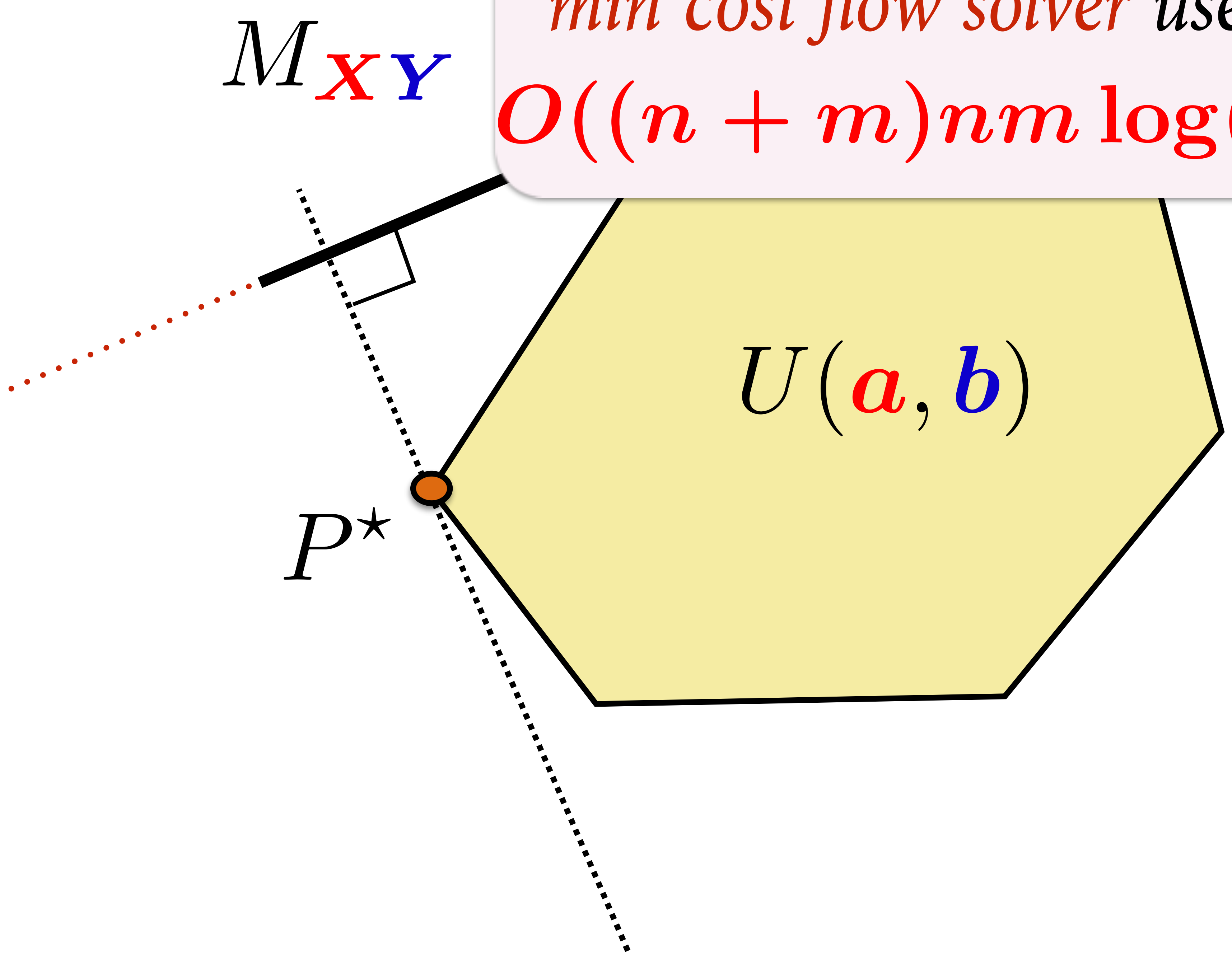
$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \max_{\substack{\boldsymbol{\alpha} \in \mathbb{R}^n, \boldsymbol{\beta} \in \mathbb{R}^m \\ \boldsymbol{\alpha_i} + \boldsymbol{\beta_j} \leq D(\boldsymbol{x_i}, \boldsymbol{y_j})^p}} \alpha^T \boldsymbol{a} + \beta^T \boldsymbol{b}$$



Solving the OT Problem



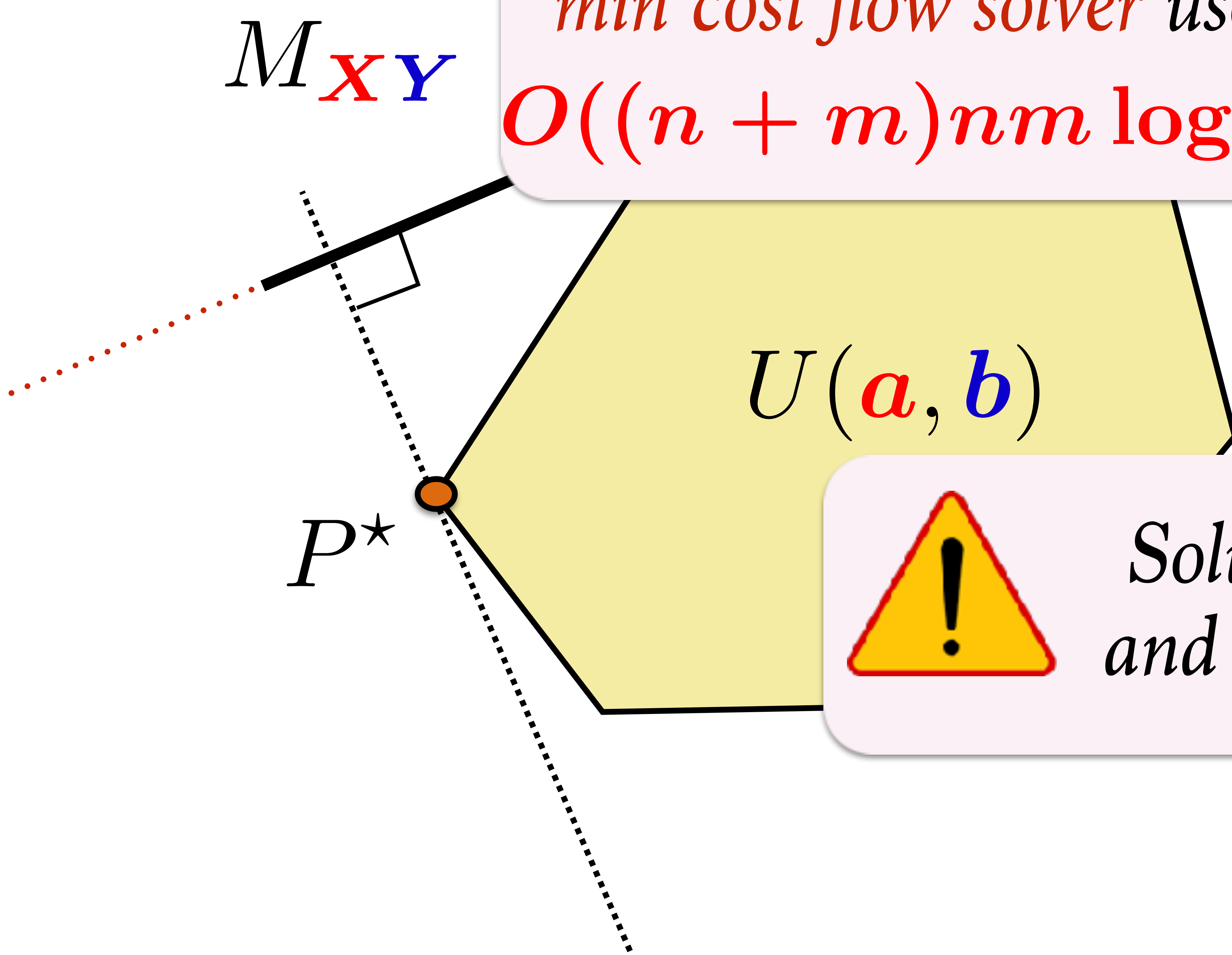
Solving the OT Problem



min cost flow solver used in practice.

$O((n+m)nm\log(n+m))$

Solving the OT Problem

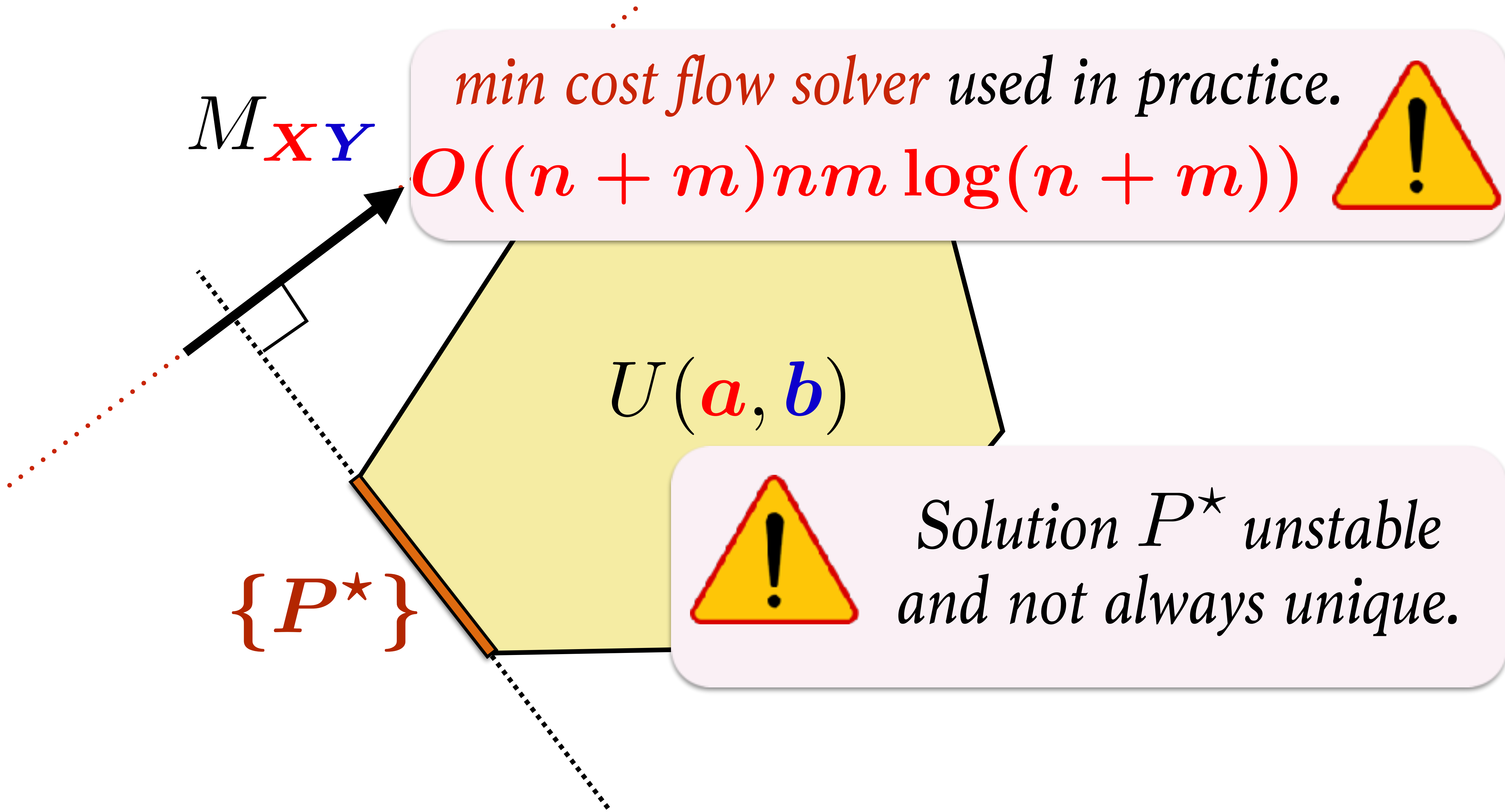


min cost flow solver used in practice.

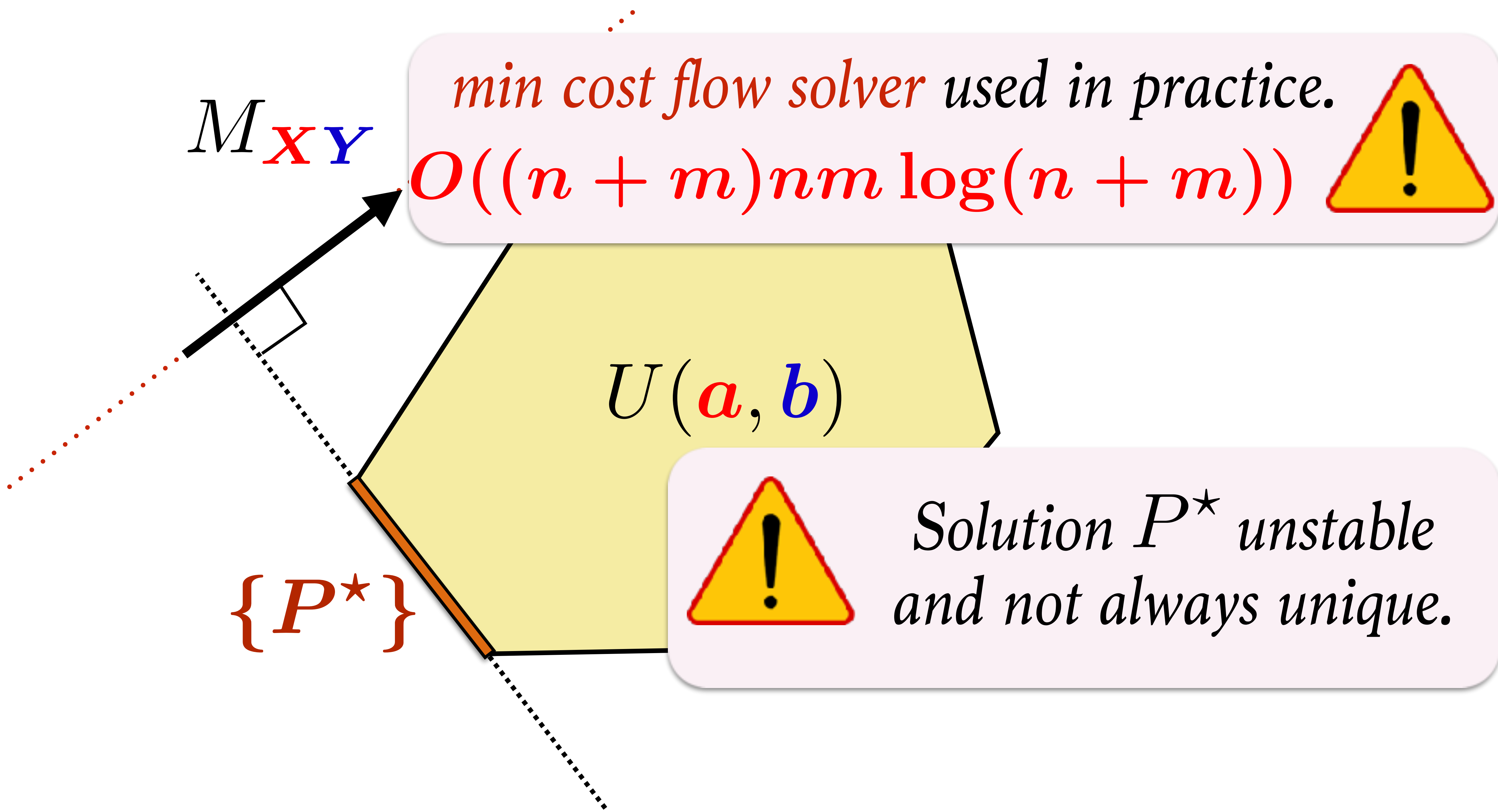
$O((n+m)nm\log(n+m))$

Solution $P^\star$ unstable and not always unique.

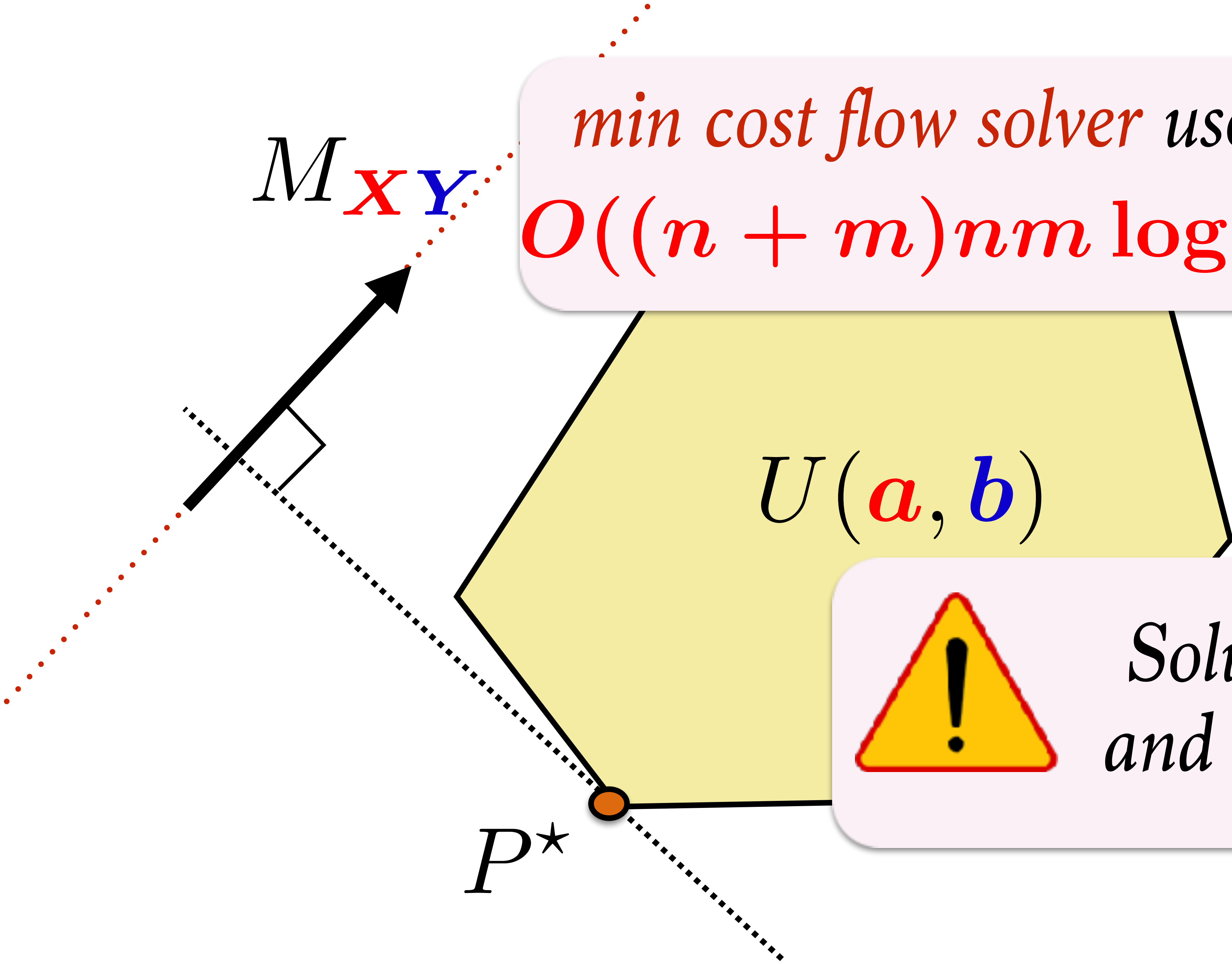
Solving the OT Problem



Solving the OT Problem



Solving the OT Problem

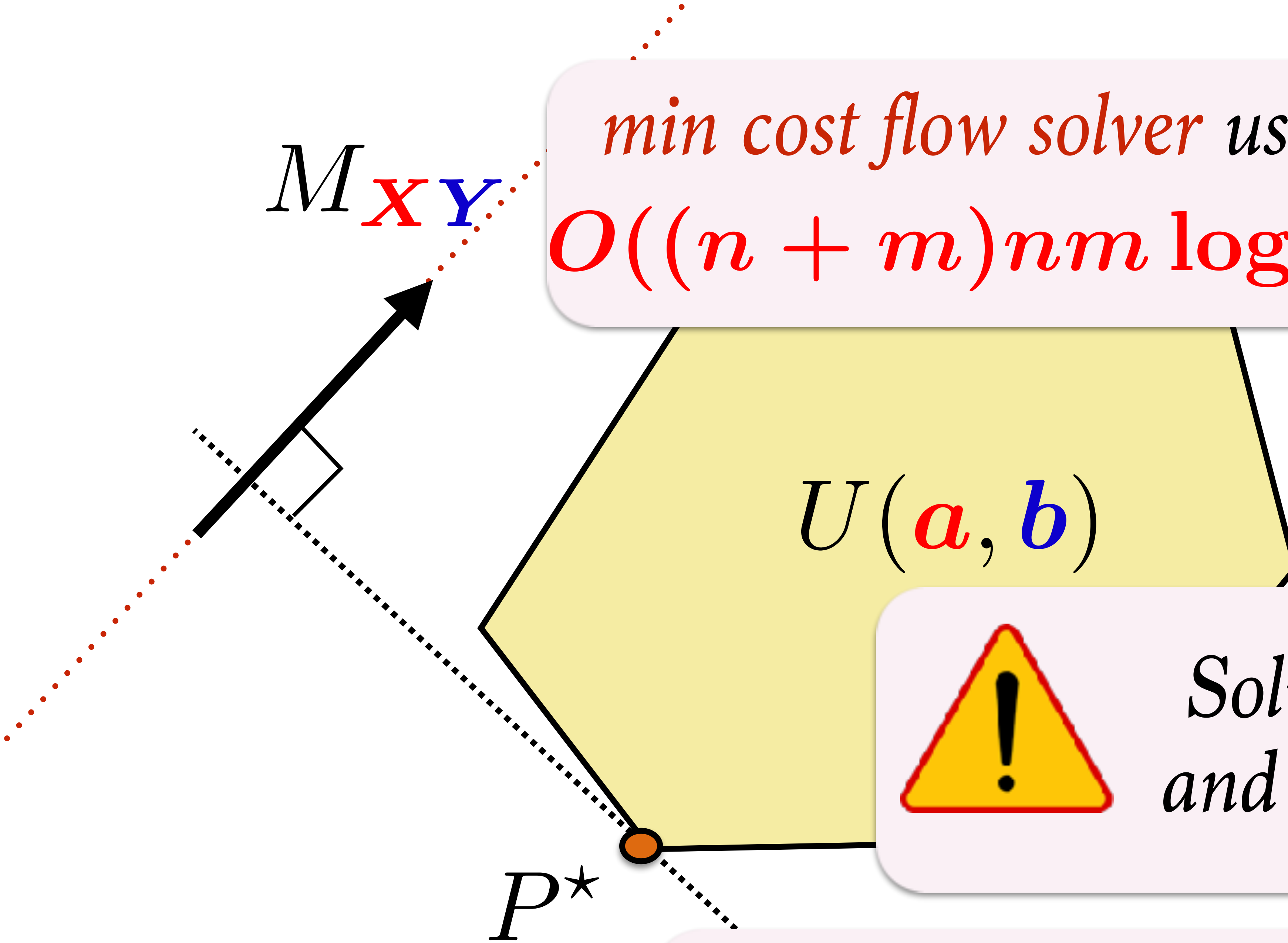


min cost flow solver used in practice.

$O((n+m)nm\log(n+m))$

Solution $P^\star$ unstable and not always unique.

Solving the OT Problem



min cost flow solver used in practice.
$O((n+m)nm\log(n+m))$

Solution $P^\star$ unstable and not always unique.

$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu})$ not differentiable.

Discrete OT Problem

emd.c

```c
/*
    emd.c

    Last update: 3/14/98

    An implementation of the Earth Movers Distance.
    Based of the solution for the Transportation problem as described in
    "Introduction to Mathematical Programming" by F. S. Hillier and
    G. J. Lieberman, McGraw-Hill, 1990.

    Copyright (C) 1998 Yossi Rubner
    Computer Science Department, Stanford University
    E-Mail: rubner@cs.stanford.edu   URL: http://vision.stanford.edu/~rubner
*/

/*#include <stdio.h>
#include <stdlib.h>*/
#include <math.h>

#include "emd.h"

#define DEBUG_LEVEL 0
/*
 DEBUG_LEVEL:
   0 = NO MESSAGES
   1 = PRINT THE NUMBER OF ITERATIONS AND THE FINAL RESULT
   2 = PRINT THE RESULT AFTER EVERY ITERATION
   3 = PRINT ALSO THE FLOW AFTER EVERY ITERATION
   4 = PRINT A LOT OF INFORMATION (PROBABLY USEFUL ONLY FOR THE AUTHOR)
*/


#define MAX_SIG_SIZE1  (MAX_SIG_SIZE+1)  /* FOR THE POSIBLE DUMMY FEATURE */

/* NEW TYPES DEFINITION */

/* node1_t IS USED FOR SINGLE-LINKED LISTS */
typedef struct node1_t {
  int i;
  double val;
  struct node1_t *Next;
} node1_t;

/* node1_t IS USED FOR DOUBLE-LINKED LISTS */
typedef struct node2_t {
  int i, j;
  double val;
  struct node2_t *NextC;               /* NEXT COLUMN */
  struct node2_t *NextR;               /* NEXT ROW */
} node2_t;



/* GLOBAL VARIABLE DECLARATION */
static int _n1, _n2;                          /* SIGNATURES SIZES */
static float _C[MAX_SIG_SIZE1][MAX_SIG_SIZE1];/* THE COST MATRIX */
static node2_t _X[MAX_SIG_SIZE1*2];            /* THE BASIC VARIABLES VECTOR */
```

Discrete OT Problem

emd.c

emd.c.6.1 <No selected symbol>

```c
/*
    emd.c

    Last update: 3/14/98

    An implementation of the Earth Movers Distance.
    Based of the solution for the Transportation problem as described in
    "Introduction to Mathematical Programming" by F. S. Hillier and
    G. J. Lieberman, McGraw-Hill, 1990.

    Copyright (C) 1998 Yossi Rubner
    Computer Science Department, Stanford University
    E-Mail: rubner@cs.stanford.edu   URL: http://vision.stanford.edu/~rubner
*/

/*#include <stdio.h>
#include <stdlib.h>*/
#include <math.h>

#include "emd.h"

#define DEBUG_LEVEL 0
/*
 DEBUG_LEVEL:
   0 = NO MESSAGES
   1 = PRINT THE NUMBER OF ITERATIONS AND THE FINAL RESULT
   2 = PRINT THE RESULT AFTER EVERY ITERATION
   3 = PRINT ALSO THE FLOW AFTER EVERY ITERATION
   4 = PRINT A LOT OF INFORMATION (PROBABLY USEFUL ONLY FOR THE AUTHOR)
*/


#define MAX_SIG_SIZE1  (MAX_SIG_SIZE+1)  /* FOR THE POSIBLE DUMMY FEATURE */

/* NEW TYPES DEFINITION */

/* node1_t IS USED FOR SINGLE-LINKED LISTS */
typedef struct node1_t {
  int i;
  double val;
  struct node1_t *Next;
} node1_t;

/* node1_t IS USED FOR DOUBLE-LINKED LISTS */
typedef struct node2_t {
  int i, j;
  double val;
  struct node2_t *NextC;               /* NEXT COLUMN */
  struct node2_t *NextR;               /* NEXT ROW */
} node2_t;



/* GLOBAL VARIABLE DECLARATION */
static int _n1, _n2;                          /* SIGNATURES SIZES */
static float _C[MAX_SIG_SIZE1][MAX_SIG_SIZE1];/* THE COST MATRIX */
static node2_t _X[MAX_SIG_SIZE1*2];            /* THE BASIC VARIABLES VECTOR */
```

Discrete OT Problem

```c
/*
    emd.c

    Last update: 3/14/98

    An implementation of the Earth Movers Distance.
    Based of the solution for the Transportation problem as described in
    "Introduction to Mathematical Programming" by F. S. Hillier and
    G. J. Lieberman, McGraw-Hill, 1990.

    Copyright (C) 1998 Yossi Rubner
    Computer Science Department, Stanford University
    E-Mail: rubner@cs.stanford.edu   URL: http://vision.stanford.edu/~rubner
*/

/*#include <stdio.h>
#include <stdlib.h>*/
#include <math.h>

#include "emd.h"

#define DEBUG_LEVEL 0
/*
 DEBUG_LEVEL:
   0 = NO MESSAGES
   1 = PRINT THE NUMBER OF ITERATIONS AND THE FINAL RESULT
   2 = PRINT THE RESULT AFTER EVERY ITERATION
   3 = PRINT ALSO THE FLOW AFTER EVERY ITERATION
   4 = PRINT A LOT OF INFORMATION (PROBABLY USEFUL ONLY FOR THE AUTHOR)
*/


#define MAX_SIG_SIZE1  (MAX_SIG_SIZE+1)  /* FOR THE POSIBLE DUMMY FEATURE */

/* NEW TYPES DEFINITION */

/* node1_t IS USED FOR SINGLE-LINKED LISTS */
typedef struct node1_t {
  int i;
  double val;
  struct node1_t *Next;
} node1_t;

/* node1_t IS USED FOR DOUBLE-LINKED LISTS */
typedef struct node2_t {
  int i, j;
  double val;
  struct node2_t *NextC;               /* NEXT COLUMN */
  struct node2_t *NextR;               /* NEXT ROW */
} node2_t;



/* GLOBAL VARIABLE DECLARATION */
static int _n1, _n2;                          /* SIGNATURES SIZES */
static float _C[MAX_SIG_SIZE1][MAX_SIG_SIZE1];/* THE COST MATRIX */
static node2_t _X[MAX_SIG_SIZE1*2];            /* THE BASIC VARIABLES VECTOR */
```

ice.

3. Computing OT for data sciences

- On an important misunderstanding
- Regularizations
- Entropic regularization
- Subspace based regularization

What matters for practitioners?

i.i.d samples $\boldsymbol{x_1, \dots, x_n} \sim \boldsymbol{\mu}$, $\boldsymbol{y_1, \dots, y_m} \sim \boldsymbol{\nu}$,

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{n}} \stackrel{\text{def}}{=} \frac{1}{n} \sum_i \delta_{\boldsymbol{x_i}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_j \delta_{\boldsymbol{y_j}}$$

Computational properties

Compute/approximate $\boldsymbol{W_p}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}})$?

Statistical properties

$$\mathbb{E}\left[\left|\boldsymbol{W_p}(\boldsymbol{\mu}, \boldsymbol{\nu}) - \boldsymbol{W_p}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}})\right|\right] \leq f(n, m)?$$

What matters for practitioners?

i.i.d samples $\boldsymbol{x_1, \dots, x_n} \sim \boldsymbol{\mu}$, $\boldsymbol{y_1, \dots, y_m} \sim \boldsymbol{\nu}$,

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{n}} \stackrel{\text{def}}{=} \frac{1}{n} \sum_i \delta_{\boldsymbol{x_i}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_j \delta_{\boldsymbol{y_j}}$$

Computational properties

$$\boldsymbol{O((n+m)nm\log(n+m))}$$

Statistical properties

$$\mathbb{E}\left[|\boldsymbol{W_p}(\boldsymbol{\mu}, \boldsymbol{\nu}) - \boldsymbol{W_p}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}})|\right] \leq f(n,m)?$$

Sample Complexity

$$\text{If } \Omega = \mathbb{R}^d, d > 3$$

$$\mathbb{E}\left[|\boldsymbol{W_p}(\boldsymbol{\mu}, \boldsymbol{\nu}) - \boldsymbol{W_p}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{n}})|\right] = O(n^{-1/d})$$

- **[Dudley'69][Dereich+'11][Fournier+'13]** & *others..*
- **[Weed/Bach'17]:** sharper results when measures' support has "low effective *d*" in metric spaces
- **[Weed/Berthet'19]** for smooth densities
- **Lower bounds:** optimal quantization error.

From theory to practice ?

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{n}} \stackrel{\text{def}}{=} \frac{1}{n}\sum_i \delta_{\boldsymbol{x_i}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}} \stackrel{\text{def}}{=} \frac{1}{m}\sum_j \delta_{\boldsymbol{y_j}}$$

Computational properties

$$O((n+m)nm\log(n+m))$$

Statistical properties

$$\mathbb{E}\left[|\boldsymbol{W_p}(\boldsymbol{\mu},\boldsymbol{\nu}) - \boldsymbol{W_p}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}},\hat{\boldsymbol{\nu}}_{\boldsymbol{n}})|\right] = O(n^{-1/d})$$

From theory to practice ?

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{n}} \stackrel{\text{def}}{=} \frac{1}{n}\sum_i \delta_{\boldsymbol{x_i}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}} \stackrel{\text{def}}{=} \frac{1}{m}\sum_j \delta_{\boldsymbol{y_j}}$$

Computational properties

$$O((n+m)nm\log(n+m))$$

Statistical properties

$$\mathbb{E}\left[|\boldsymbol{W_p}(\boldsymbol{\mu},\boldsymbol{\nu}) - \boldsymbol{W_p}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}},\hat{\boldsymbol{\nu}}_{\boldsymbol{n}})|\right] = O(n^{-1/d})$$

In data sciences we *must* regularize the problem to improve on either (or both) aspects!

Many ways to regularize (dual) OT

$$\sup_{\boldsymbol{\varphi}(x)+\boldsymbol{\psi}(y)\leq \boldsymbol{c}(x,y)} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu}.$$

Many ways to regularize (dual) OT

$$\sup_{\varphi(x)+\psi(y)\leq c(x,y)} \int \varphi d\mu + \int \psi d\nu .$$

- RKHS for potentials **[GCBP'16]**, dualize/smooth indicator constraint

Many ways to regularize (dual) OT

$$\sup_{\varphi(x)+\psi(y)\leq c(x,y)} \int \varphi d\mu + \int \psi d\nu.$$

- RKHS for potentials **[GCBP'16]**, dualize/smooth indicator constraint

$$W_1(\mu,\nu) = \sup_{\varphi \text{ 1-Lipschitz}} \int \varphi(d\mu - d\nu).$$

- Parameterize functions using ReLU Deep net with bounded weights **[Arjovsky+'17]** or use Wavelet decompositions **[Shirdonkhar+'08]** for low *d*.

Many ways to regularize (primal) OT

$$\inf_{\boldsymbol{P} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \iint \boldsymbol{c}(x, y) \boldsymbol{P}(dx, dy).$$

Many ways to regularize (primal) OT

$$\inf_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \iint \boxed{\boldsymbol{c}(x,y)}\,\boldsymbol{P}(dx,dy).$$

- Change cost function: threshold metric **[Pele+'09]**, use geodesic distance on graphs **[Beckman'52]** **[Lin+'07]**, **[Solomon+'14]**, simplifies the LP.

Many ways to regularize (primal) OT

$$\inf_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \iint \boldsymbol{c}(x,y)\boldsymbol{P}(dx,dy).$$

- Change cost function: threshold metric **[Pele+'09]**, use geodesic distance on graphs **[Beckman'52]** **[Lin+'07]**, **[Solomon+'14]**, simplifies the LP.

- Change measures: Quantize first **[Canas+'12]**; use Gaussians **[Gelbrich'92]**; projections on random lines **[Rabin+'11]** & k-dimensional subspaces **[Paty+'19][Weed+'19]**.

Many ways to regularize (primal) OT

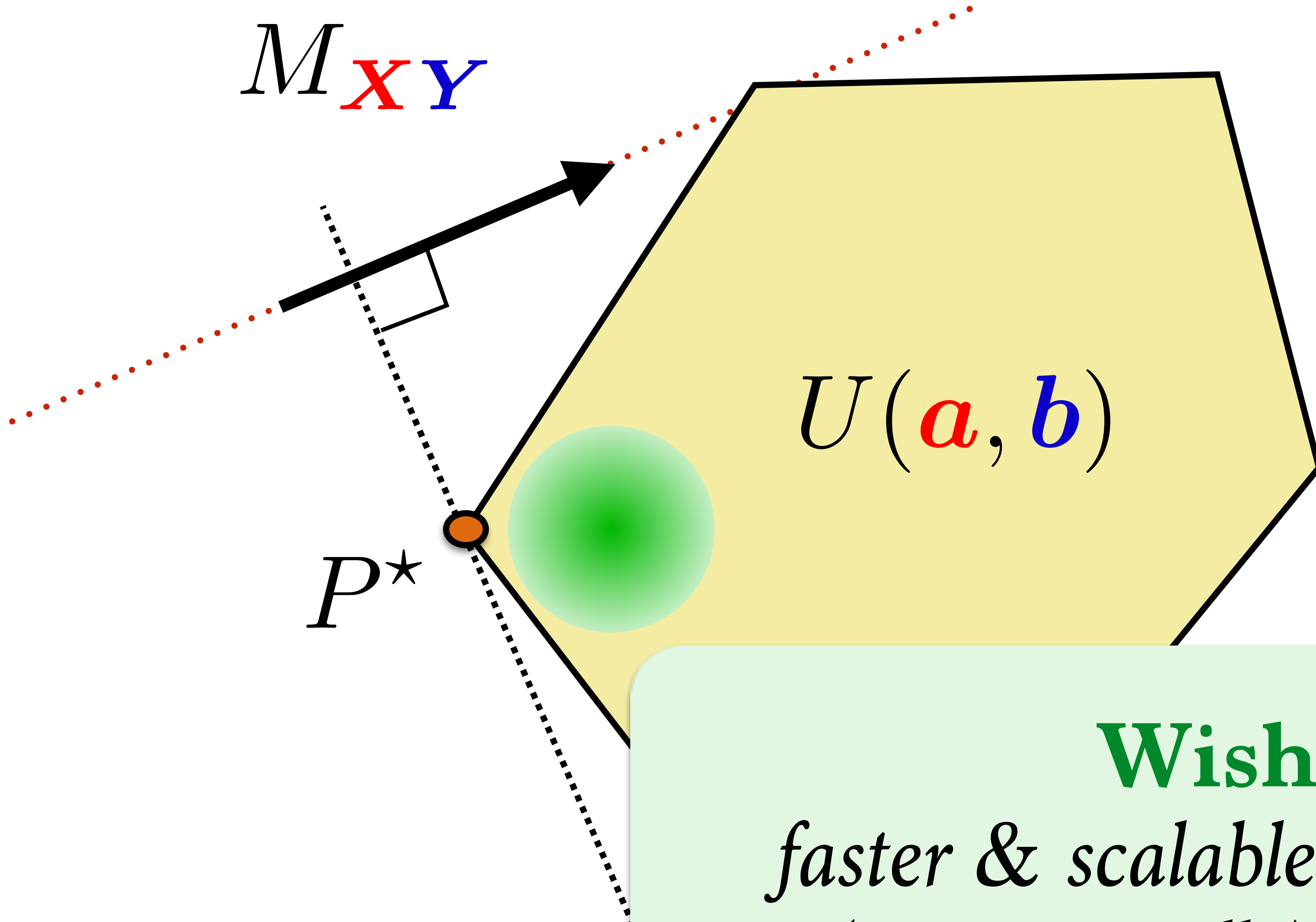
$$\inf_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \iint \boldsymbol{c}(x,y)\boldsymbol{P}(dx,dy).$$

- Change cost function: threshold metric **[Pele+'09]**, use geodesic distance on graphs **[Beckman'52]** **[Lin+'07]**, **[Solomon+'14]**, simplifies the LP.

- Change measures: Quantize first **[Canas+'12]**; use Gaussians **[Gelbrich'92]**; projections on random lines **[Rabin+'11]** & k-dimensional subspaces **[Paty+'19][Weed+'19]**.

- Add prior on coupling, e.g. (entropic) regularization **[C'13][GP'16][GCBP'16] [GCBCP'19]**

Regularization on the Primal



Wishlist:

faster & *scalable*, more *stable*,
(automatically) differentiable

Entropic Regularization [**Wilson'62**]

Def. Regularized Wasserstein, $\gamma \geq 0$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \min_{\boldsymbol{P} \in U(\boldsymbol{a}, \boldsymbol{b})} \langle \boldsymbol{P}, M_{\boldsymbol{XY}} \rangle - \gamma E(\boldsymbol{P})$$

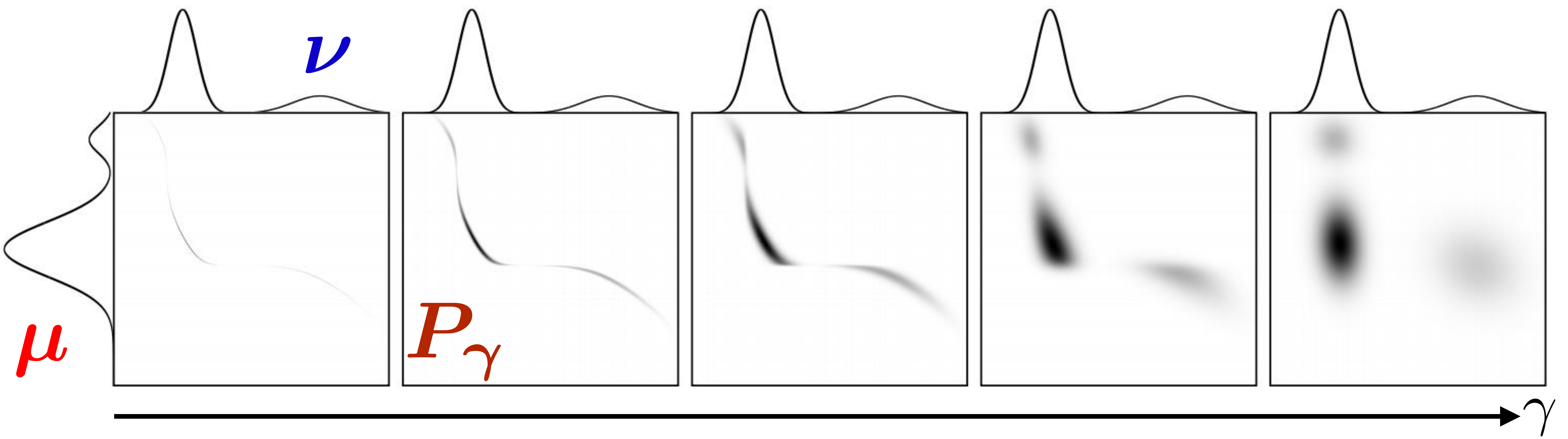
$$E(P) \stackrel{\text{def}}{=} - \sum_{i,j=1}^{nm} P_{ij}(\log P_{ij} - 1)$$

Note: Unique optimal solution because of strong concavity of entropy

Entropic Regularization [**Wilson'62**]

Def. Regularized Wasserstein, $\gamma \geq 0$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \min_{\boldsymbol{P} \in U(\boldsymbol{a}, \boldsymbol{b})} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle - \gamma E(\boldsymbol{P})$$

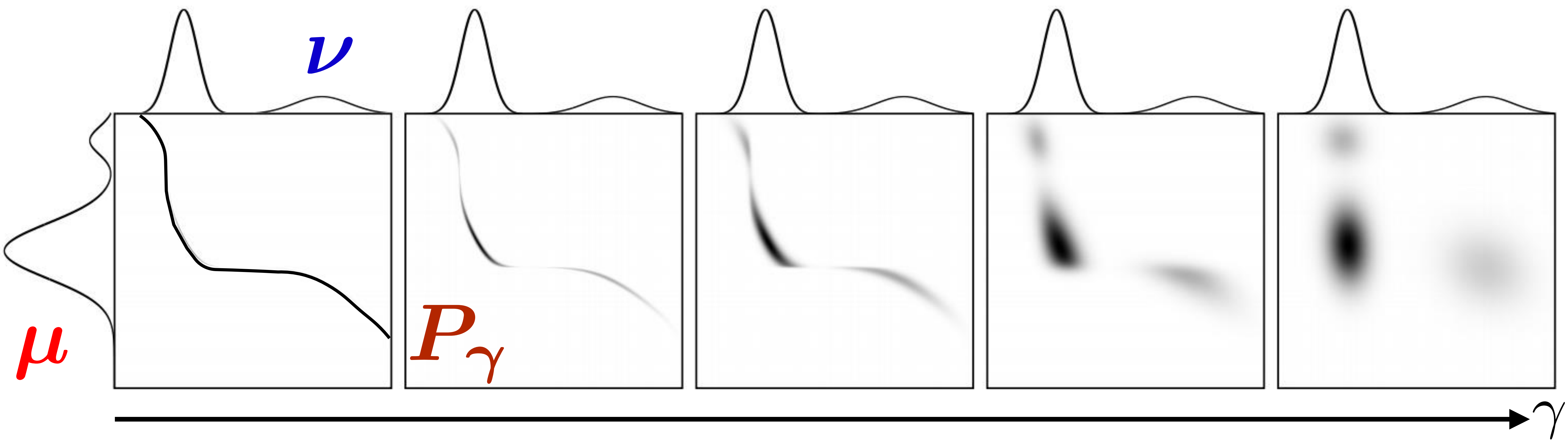


Note: Unique optimal solution because of strong concavity of entropy

Entropic Regularization [Wilson'62]

Def. Regularized Wasserstein, $\gamma \geq 0$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \min_{\boldsymbol{P} \in U(\boldsymbol{a}, \boldsymbol{b})} \langle \boldsymbol{P}, M_{\boldsymbol{XY}} \rangle - \gamma E(\boldsymbol{P})$$

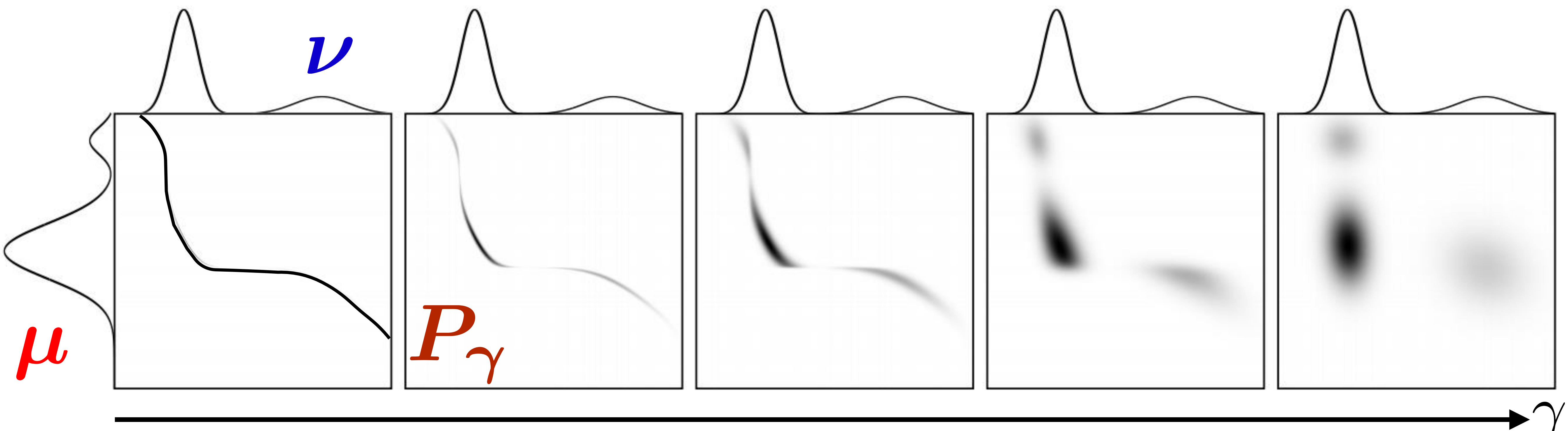


Note: Unique optimal solution because of strong concavity of entropy

Entropic Regularization [Wilson'62]

Def. Regularized Wasserstein, $\gamma \geq 0$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \min_{\boldsymbol{P} \in U(\boldsymbol{a}, \boldsymbol{b})} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle - \gamma E(\boldsymbol{P})$$



$\approx$ "$\boldsymbol{y} = \boldsymbol{T}(\boldsymbol{x})$"

Note: Unique optimal solution because of strong concavity of entropy

Fast & Scalable Algorithm

Prop. If $P_\gamma \overset{\text{def}}{=} \underset{\boldsymbol{P} \in U(\boldsymbol{a},\boldsymbol{b})}{\operatorname{argmin}} \langle \boldsymbol{P}, M_{\boldsymbol{X}\boldsymbol{Y}} \rangle - \gamma E(\boldsymbol{P})$

then $\exists! \boldsymbol{u} \in \mathbb{R}_+^n, \boldsymbol{v} \in \mathbb{R}_+^m$, such that

$$P_\gamma = \mathbf{diag}(\boldsymbol{u}) K \mathbf{diag}(\boldsymbol{v}), \quad K \overset{\text{def}}{=} e^{-M_{\boldsymbol{X}\boldsymbol{Y}}/\gamma}$$

Fast & Scalable Algorithm

Prop. If $P_\gamma \stackrel{\text{def}}{=} \underset{\boldsymbol{P}\in U(\boldsymbol{a},\boldsymbol{b})}{\operatorname{argmin}} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle - \gamma E(\boldsymbol{P})$

then $\exists! \boldsymbol{u} \in \mathbb{R}_+^n, \boldsymbol{v} \in \mathbb{R}_+^m$, such that

$$P_\gamma = \mathbf{diag}(\boldsymbol{u}) K \mathbf{diag}(\boldsymbol{v}), \quad K \stackrel{\text{def}}{=} e^{-M_{\mathbf{XY}}/\gamma}$$

$$L(P,\alpha,\beta) = \sum_{ij} P_{ij}M_{ij} + \gamma P_{ij}(\log P_{ij} - 1) + \alpha^T(P\mathbf{1} - \boldsymbol{a}) + \beta^T(P^T\mathbf{1} - \boldsymbol{b})$$

$$\partial L/\partial P_{ij} = M_{ij} + \gamma \log P_{ij} + \alpha_i + \beta_j$$

$$(\boldsymbol{\partial L/\partial P_{ij}} = 0) \Rightarrow P_{ij} = \boldsymbol{e}^{\frac{\boldsymbol{\alpha_i}}{\gamma}}\ e^{-\frac{M_{ij}}{\gamma}}\ \boldsymbol{e}^{\frac{\boldsymbol{\beta_j}}{\gamma}} = \boldsymbol{u_i}\ K_{ij} \boldsymbol{v_j}$$

Fast & Scalable Algorithm

Prop. If $P_\gamma \stackrel{\text{def}}{=} \underset{\boldsymbol{P} \in U(\boldsymbol{a},\boldsymbol{b})}{\operatorname{argmin}} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle - \gamma E(\boldsymbol{P})$ then $\exists ! \boldsymbol{u} \in \mathbb{R}_+^n, \boldsymbol{v} \in \mathbb{R}_+^m$, such that

$$P_\gamma = \mathbf{diag}(\boldsymbol{u}) K \mathbf{diag}(\boldsymbol{v}), \quad K \stackrel{\text{def}}{=} e^{-M_{\mathbf{XY}}/\gamma}$$

$$P_\gamma \in U(\boldsymbol{a}, \boldsymbol{b}) \Leftrightarrow \begin{cases} \mathbf{diag}(\boldsymbol{u}) K \mathbf{diag}(\boldsymbol{v}) \mathbf{1}_m & = \boldsymbol{a} \\ \mathbf{diag}(\boldsymbol{v}) K^T \mathbf{diag}(\boldsymbol{u}) \mathbf{1}_n & = \boldsymbol{b} \end{cases}$$

Fast & Scalable Algorithm

Prop. If $P_\gamma \stackrel{\text{def}}{=} \operatorname*{argmin}_{\mathbf{P} \in U(\mathbf{a},\mathbf{b})} \langle \mathbf{P}, M_{\mathbf{XY}} \rangle - \gamma E(\mathbf{P})$

then $\exists! \mathbf{u} \in \mathbb{R}_+^n, \mathbf{v} \in \mathbb{R}_+^m$, such that

$$P_\gamma = \mathbf{diag}(\mathbf{u}) K \mathbf{diag}(\mathbf{v}), \quad K \stackrel{\text{def}}{=} e^{-M_{\mathbf{XY}}/\gamma}$$

$$P_\gamma \in U(\mathbf{a}, \mathbf{b}) \Leftrightarrow \begin{cases} \mathbf{diag}(\mathbf{u}) K \mathbf{diag}(\mathbf{v}) \mathbf{1}_m & = \mathbf{a} \\ \mathbf{diag}(\mathbf{v}) K^T \underbrace{\mathbf{diag}(\mathbf{u}) \mathbf{1}_n}_{\mathbf{u}} & = \mathbf{b} \end{cases}$$

Fast & Scalable Algorithm

Prop. If $P_\gamma \stackrel{\text{def}}{=} \underset{\boldsymbol{P}\in U(\boldsymbol{a},\boldsymbol{b})}{\text{argmin}} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle - \gamma E(\boldsymbol{P})$

then $\exists! \boldsymbol{u} \in \mathbb{R}^n_+, \boldsymbol{v} \in \mathbb{R}^m_+$, such that

$$P_\gamma = \mathbf{diag}(\boldsymbol{u}) K \mathbf{diag}(\boldsymbol{v}), \quad K \stackrel{\text{def}}{=} e^{-M_{\mathbf{XY}}/\gamma}$$

$$P_\gamma \in U(\boldsymbol{a}, \boldsymbol{b}) \Leftrightarrow \begin{cases} \mathbf{diag}(\boldsymbol{u}) K \overbrace{\mathbf{diag}(\boldsymbol{v}) \mathbf{1}_m}^{\boldsymbol{v}} & = \boldsymbol{a} \\ \mathbf{diag}(\boldsymbol{v}) K^T \underbrace{\mathbf{diag}(\boldsymbol{u}) \mathbf{1}_n}_{\boldsymbol{u}} & = \boldsymbol{b} \end{cases}$$

Fast & Scalable Algorithm

Prop. If $P_\gamma \stackrel{\text{def}}{=} \underset{\boldsymbol{P}\in U(\boldsymbol{a},\boldsymbol{b})}{\operatorname{argmin}} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle - \gamma E(\boldsymbol{P})$

then $\exists! \boldsymbol{u} \in \mathbb{R}_+^n, \boldsymbol{v} \in \mathbb{R}_+^m$, such that

$$P_\gamma = \mathbf{diag}(\boldsymbol{u}) K \mathbf{diag}(\boldsymbol{v}), \quad K \stackrel{\text{def}}{=} e^{-M_{\mathbf{XY}}/\gamma}$$

$$P_\gamma \in U(\boldsymbol{a}, \boldsymbol{b}) \Leftrightarrow \begin{cases} \mathbf{diag}(\boldsymbol{u}) K \boldsymbol{v} & = \boldsymbol{a} \\ \mathbf{diag}(\boldsymbol{v}) K^T \boldsymbol{u} & = \boldsymbol{b} \end{cases}$$

Fast & Scalable Algorithm

Prop. If $P_\gamma \stackrel{\text{def}}{=} \underset{\mathbf{P}\in U(\mathbf{a},\mathbf{b})}{\operatorname{argmin}} \langle \mathbf{P}, M_{\mathbf{XY}} \rangle - \gamma E(\mathbf{P})$

then $\exists! \mathbf{u} \in \mathbb{R}_+^n, \mathbf{v} \in \mathbb{R}_+^m$, such that

$$P_\gamma = \mathbf{diag}(\mathbf{u}) K \mathbf{diag}(\mathbf{v}), \quad K \stackrel{\text{def}}{=} e^{-M_{\mathbf{XY}}/\gamma}$$

$$P_\gamma \in U(\mathbf{a}, \mathbf{b}) \Leftrightarrow \begin{cases} \mathbf{u} \odot K\mathbf{v} = \mathbf{a} \\ \mathbf{v} \odot K^T\mathbf{u} = \mathbf{b} \end{cases}$$

Fast & Scalable Algorithm

Prop. If $P_\gamma \stackrel{\text{def}}{=} \underset{\boldsymbol{P}\in U(\boldsymbol{a},\boldsymbol{b})}{\operatorname{argmin}} \langle \boldsymbol{P}, M_{\boldsymbol{XY}} \rangle - \gamma E(\boldsymbol{P})$

then $\exists! \boldsymbol{u} \in \mathbb{R}^n_+, \boldsymbol{v} \in \mathbb{R}^m_+$, such that

$$P_\gamma = \mathbf{diag}(\boldsymbol{u}) K \mathbf{diag}(\boldsymbol{v}), \quad K \stackrel{\text{def}}{=} e^{-M_{\boldsymbol{XY}}/\gamma}$$

$$P_\gamma \in U(\boldsymbol{a}, \boldsymbol{b}) \Leftrightarrow \begin{cases} \boldsymbol{u} = \boldsymbol{a}/K\boldsymbol{v} \\ \boldsymbol{v} = \boldsymbol{b}/K^T\boldsymbol{u} \end{cases}$$

Fast & Scalable Algorithm

Sinkhorn's Algorithm : Repeat

1. $\boldsymbol{u} = \boldsymbol{a} / K \boldsymbol{v}$
2. $\boldsymbol{v} = \boldsymbol{b} / K^T \boldsymbol{u}$

Fast & Scalable Algorithm

Sinkhorn's Algorithm : Repeat

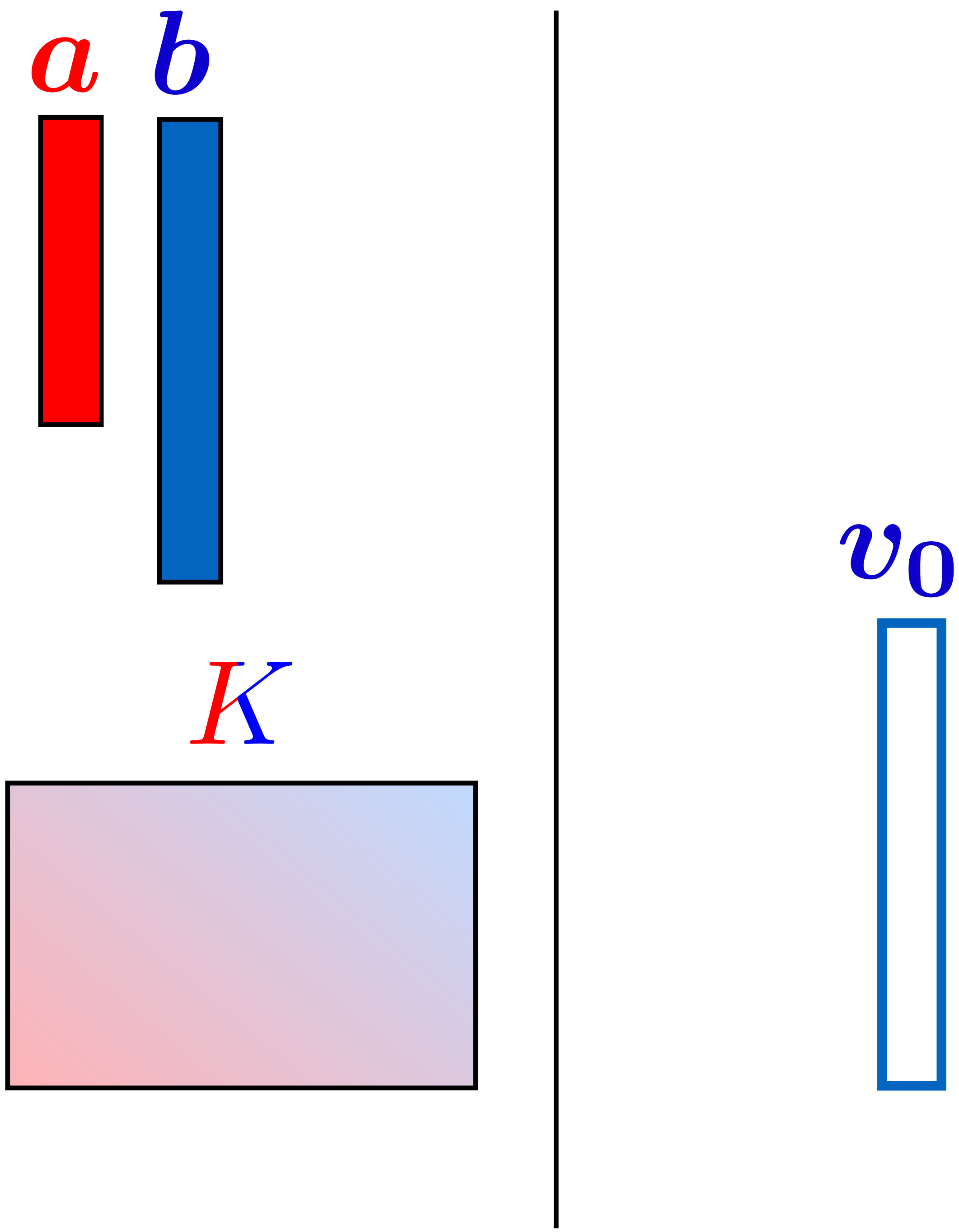
1. $\boldsymbol{u} = \boldsymbol{a} / K\boldsymbol{v}$
2. $\boldsymbol{v} = \boldsymbol{b} / K^T\boldsymbol{u}$

- [**Sinkhorn'64**] proved convergence for the first time.
- [**Lorenz'89**] linear convergence, see [**Altschuler'17**]
- $O(nm)$ complexity, GPGPU parallel [**Cuturi'13**] .
- $O(n \log n)$ on gridded spaces using convolutions. [**Solomon'15**]

Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

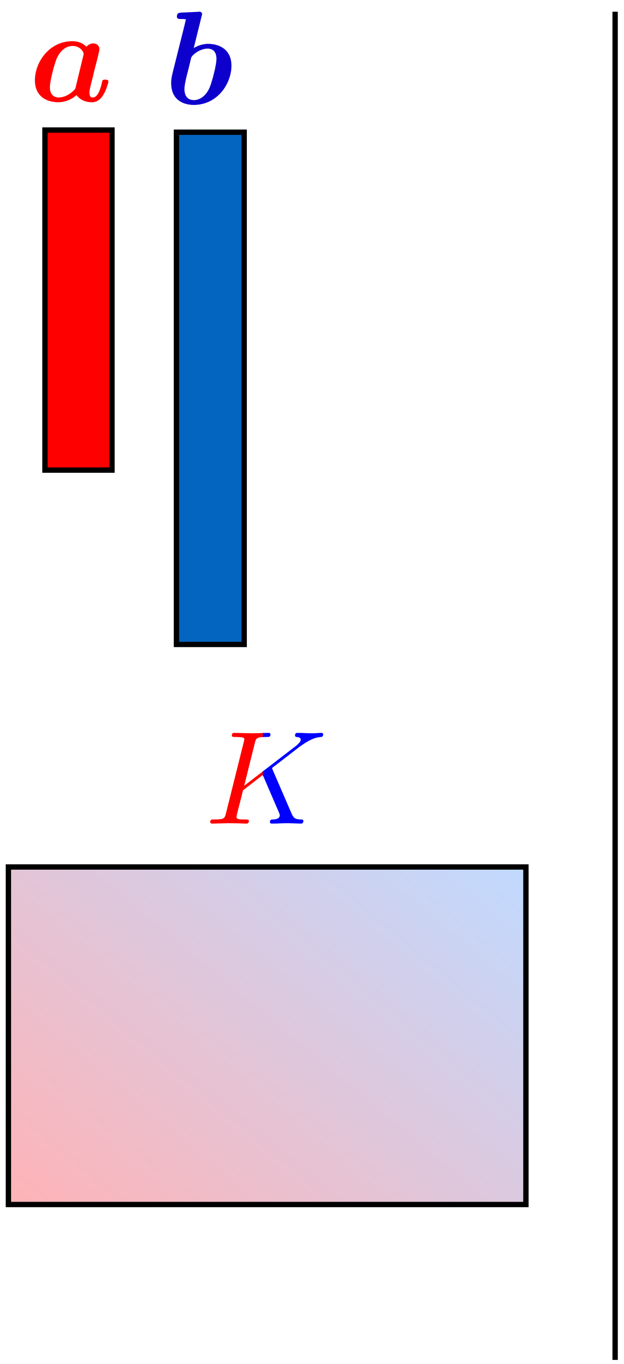
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$

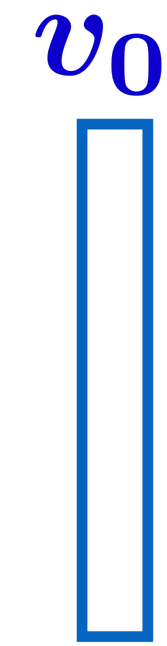


Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$

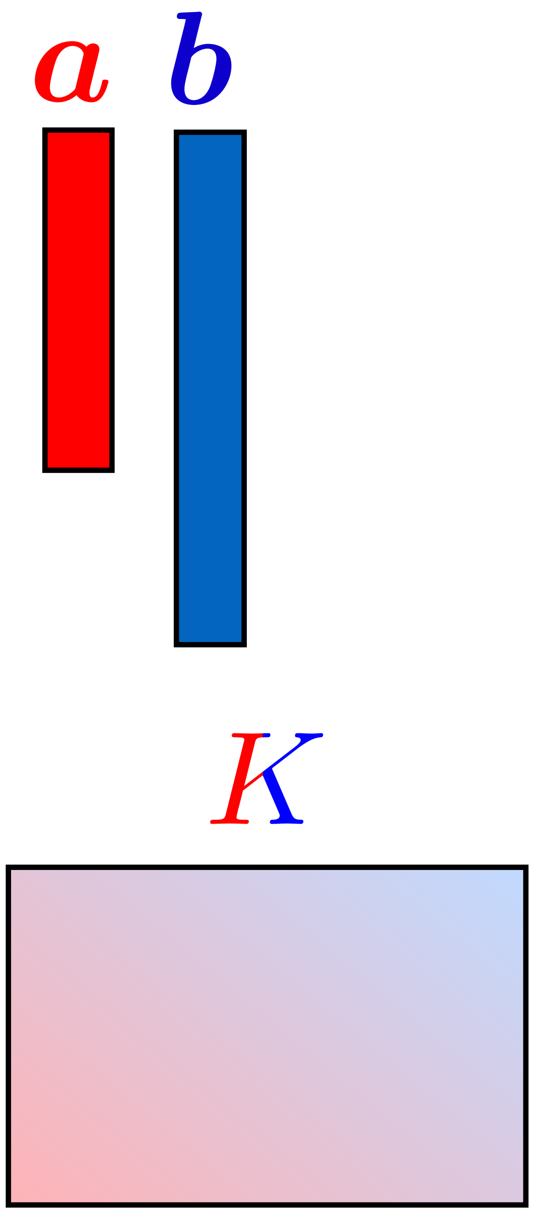


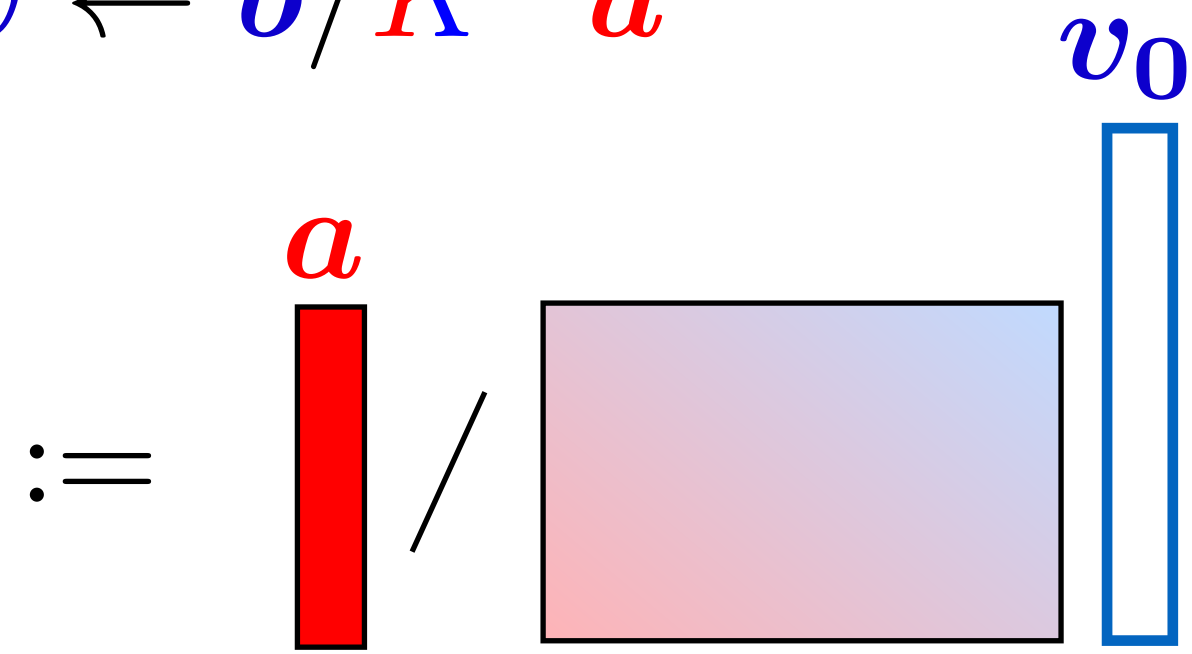


Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$

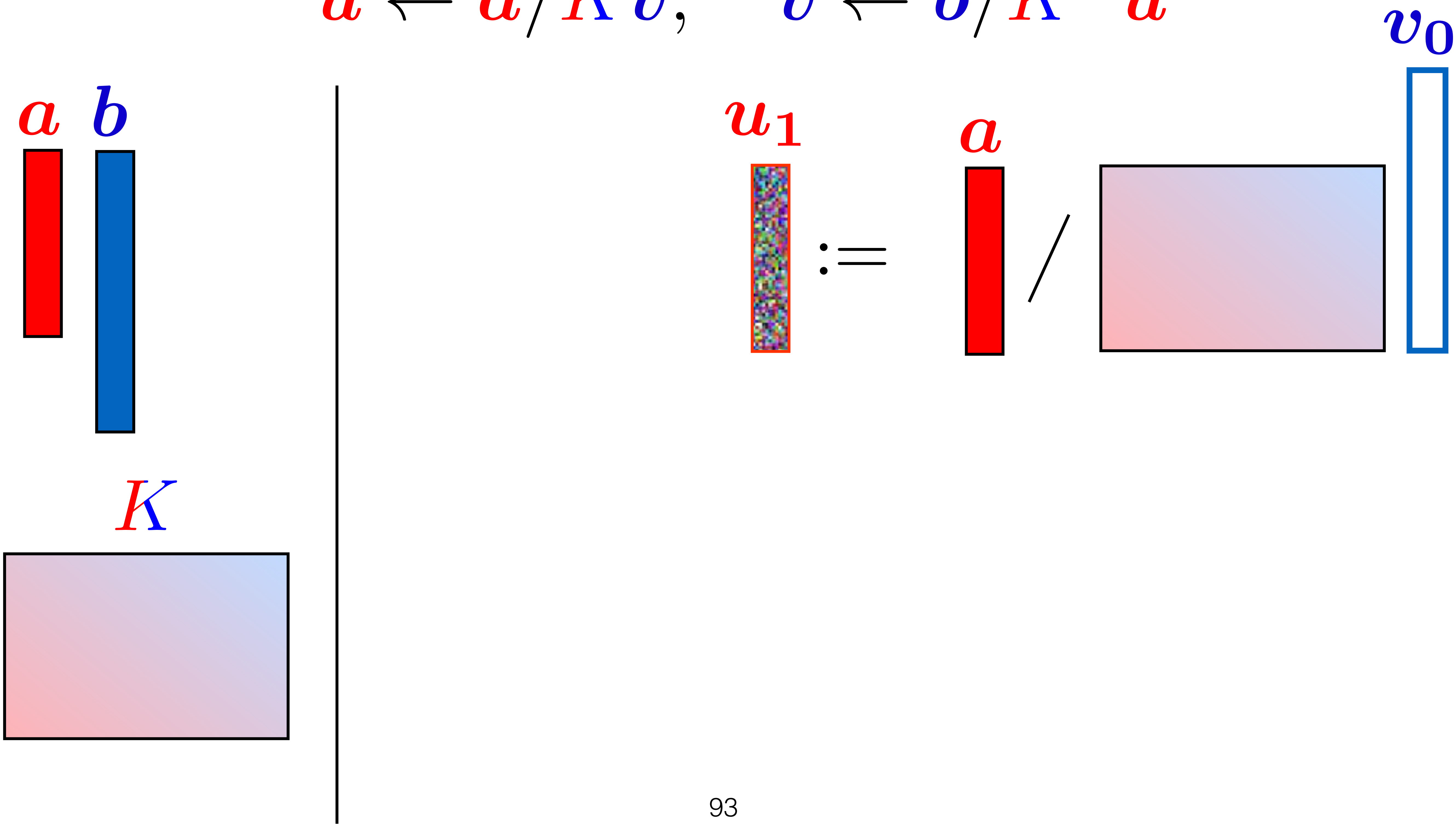




Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

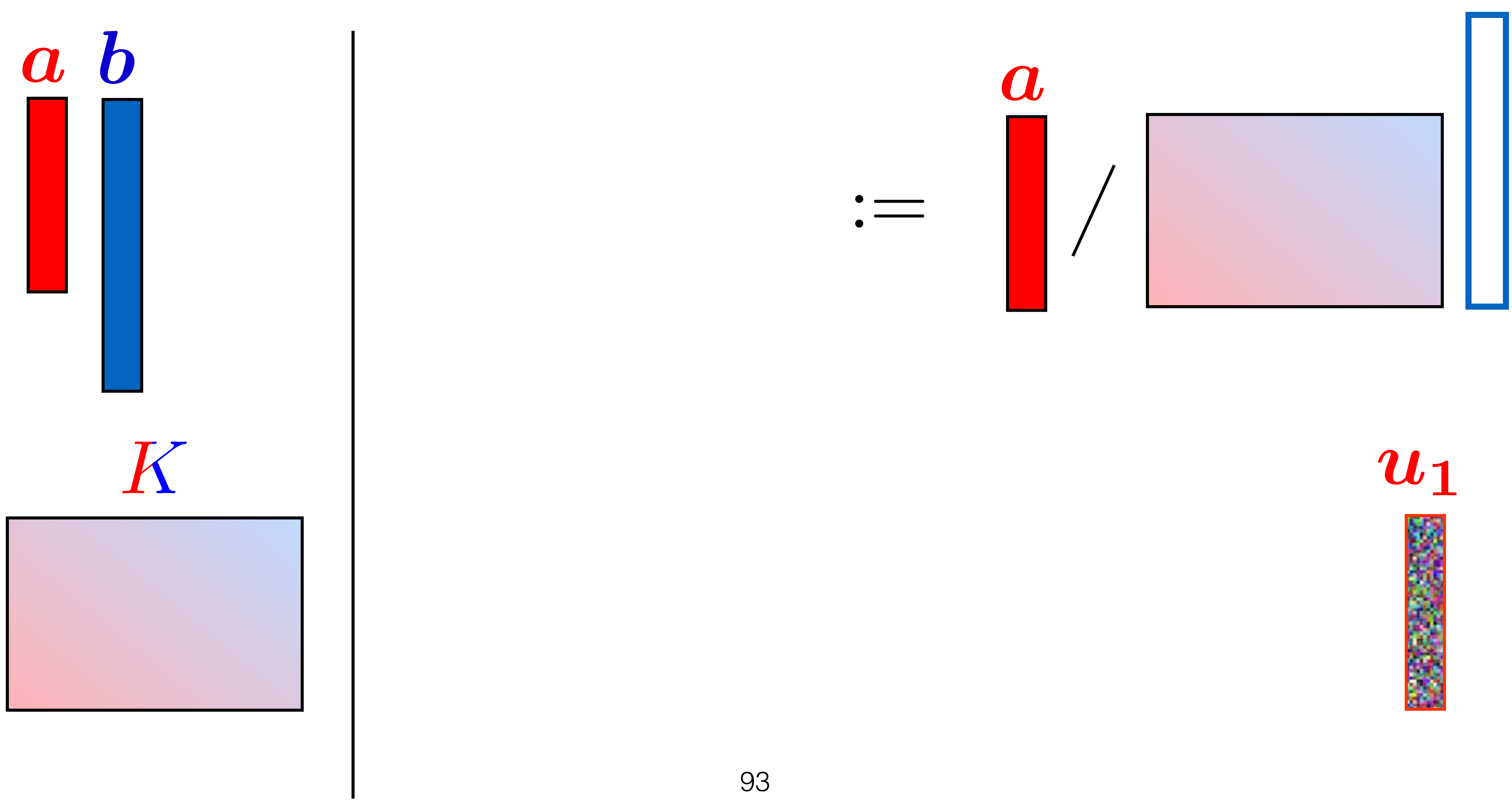
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

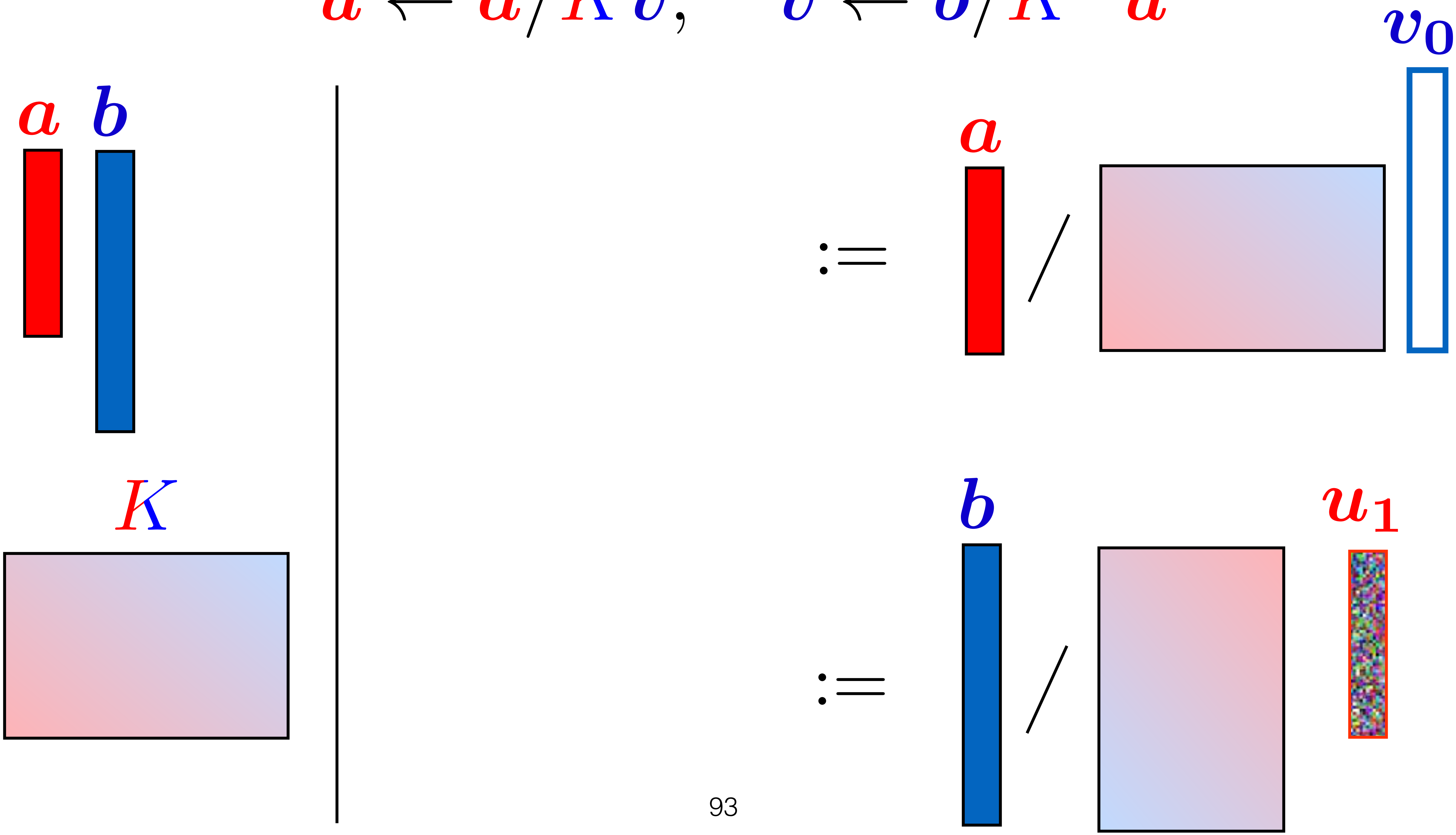
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

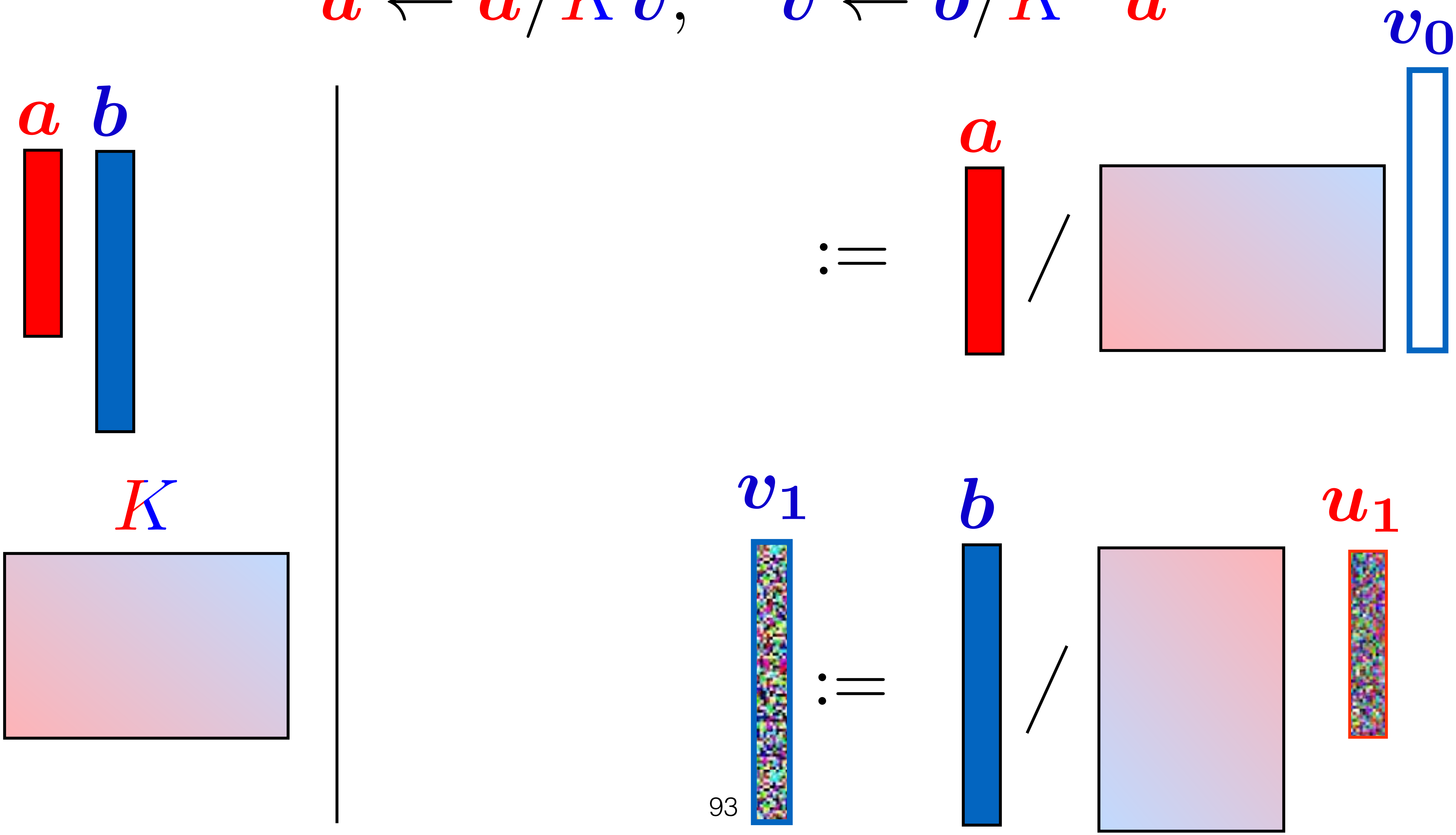
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

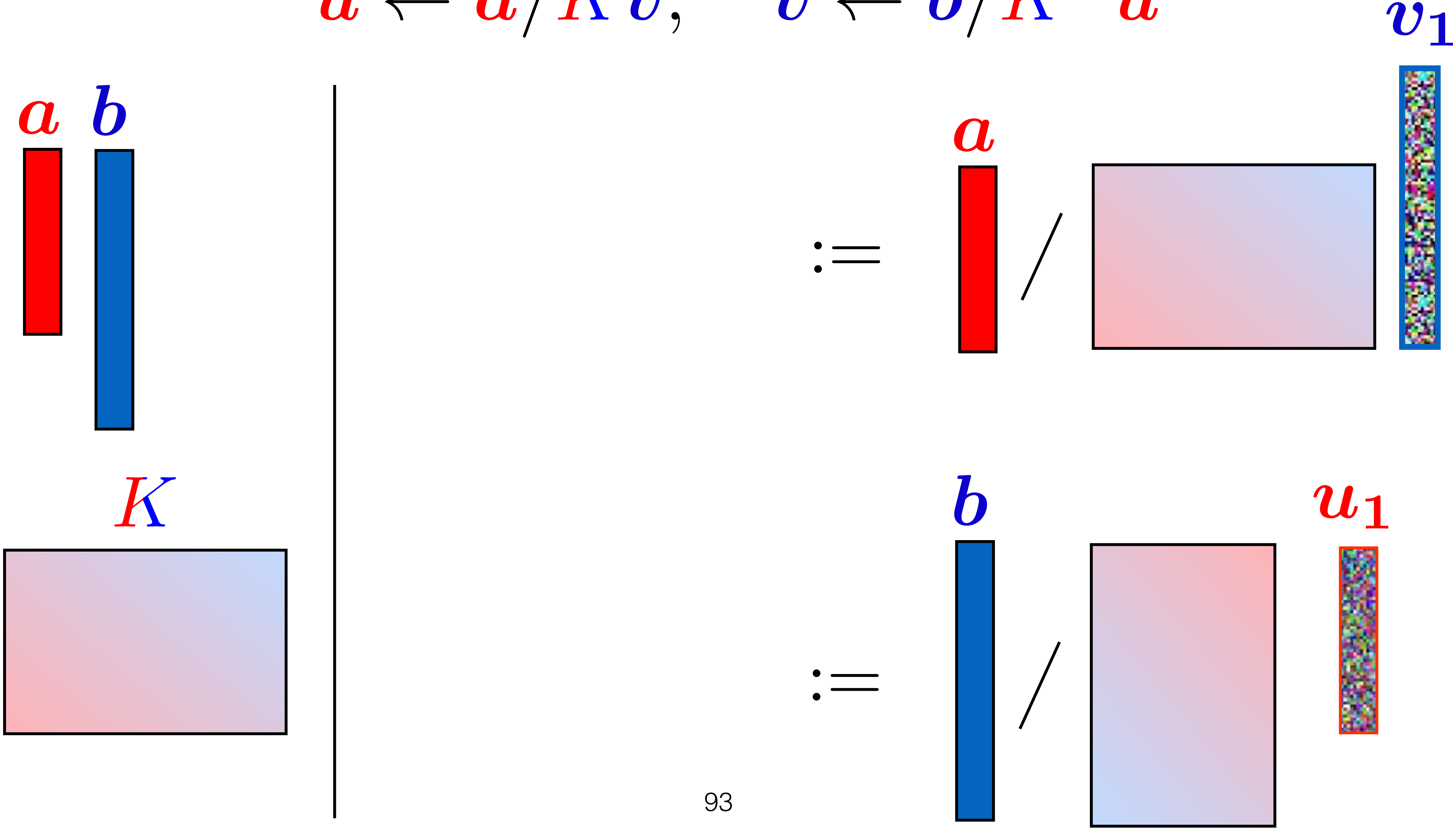
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

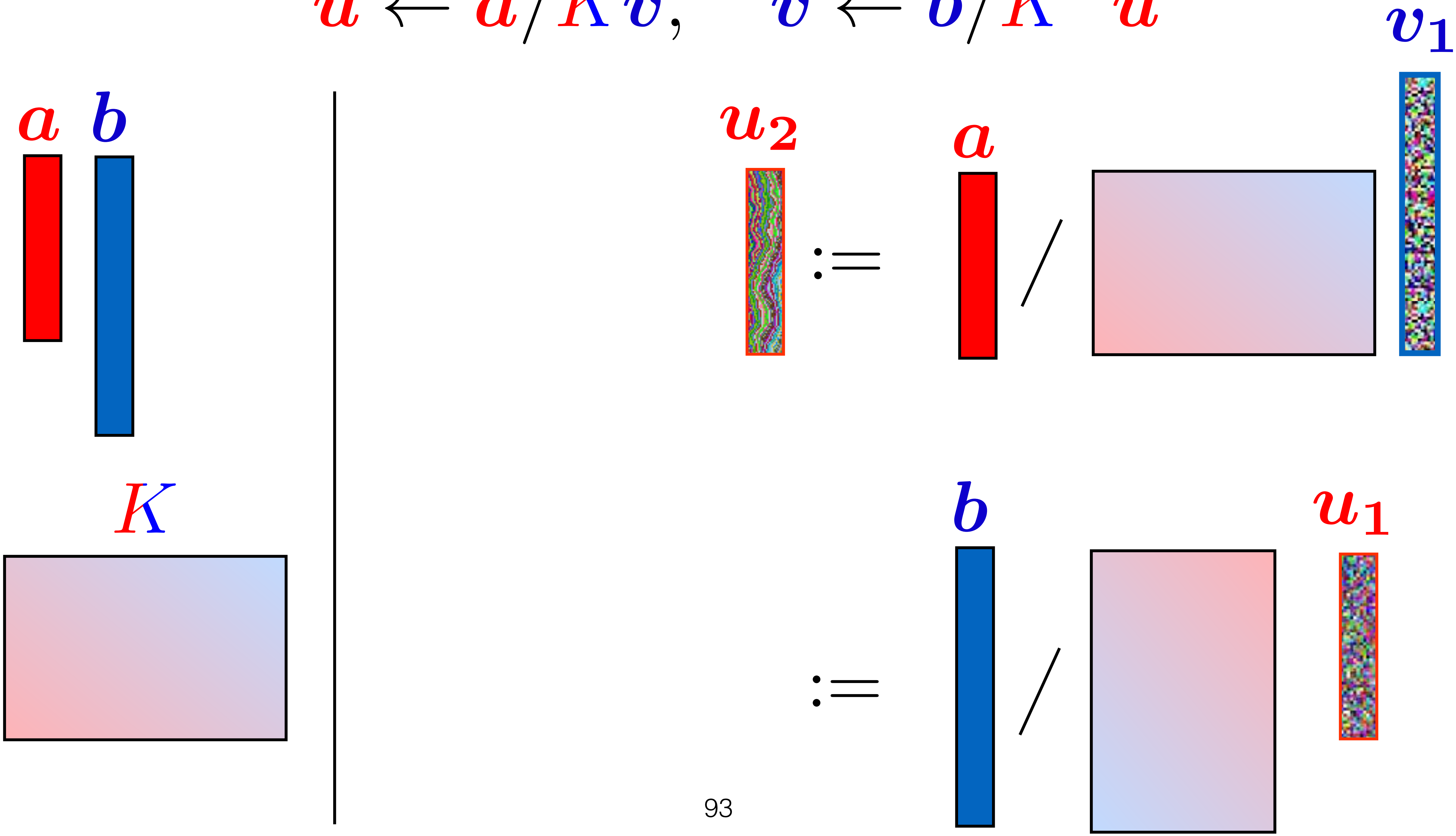
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

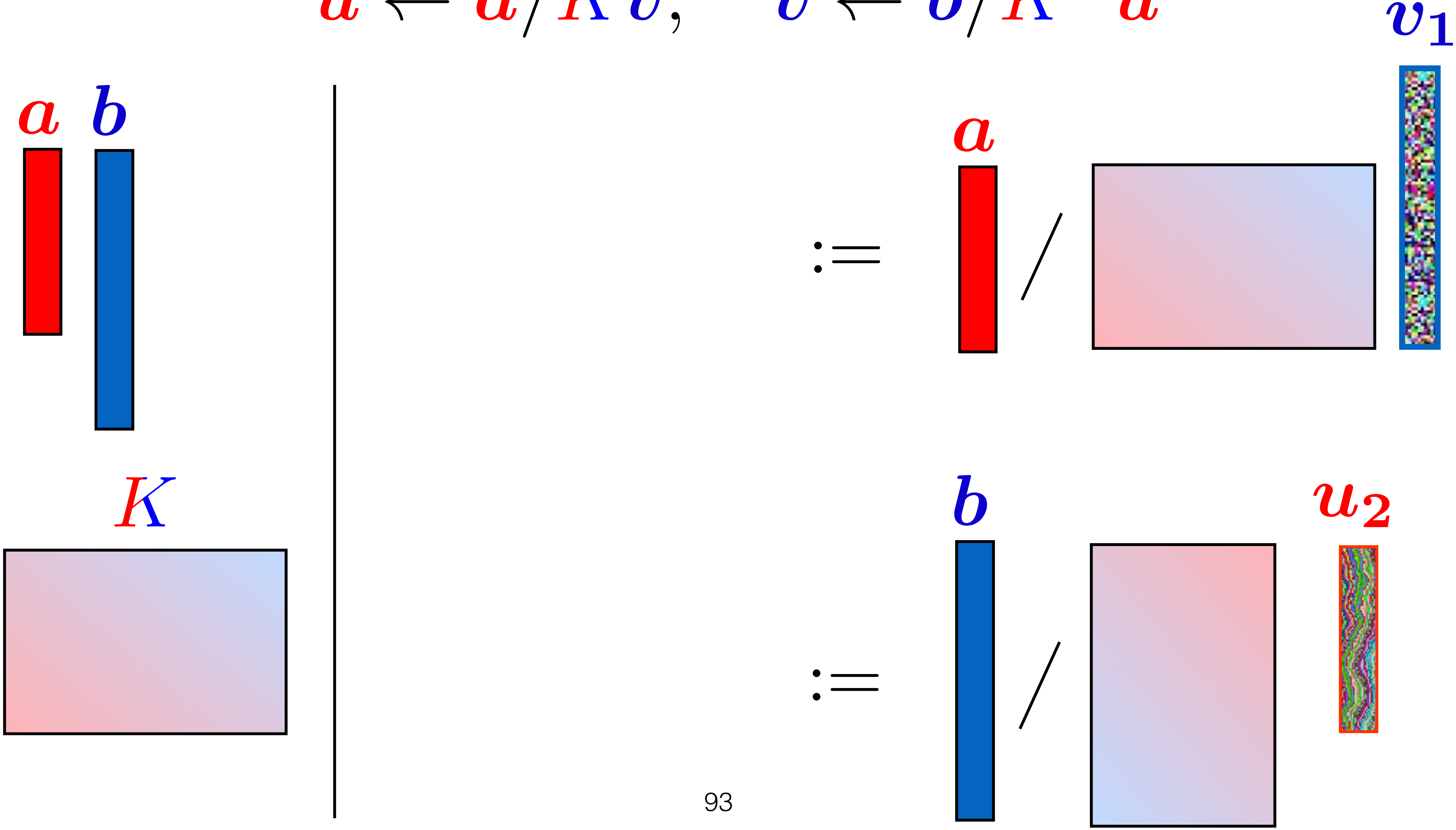
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

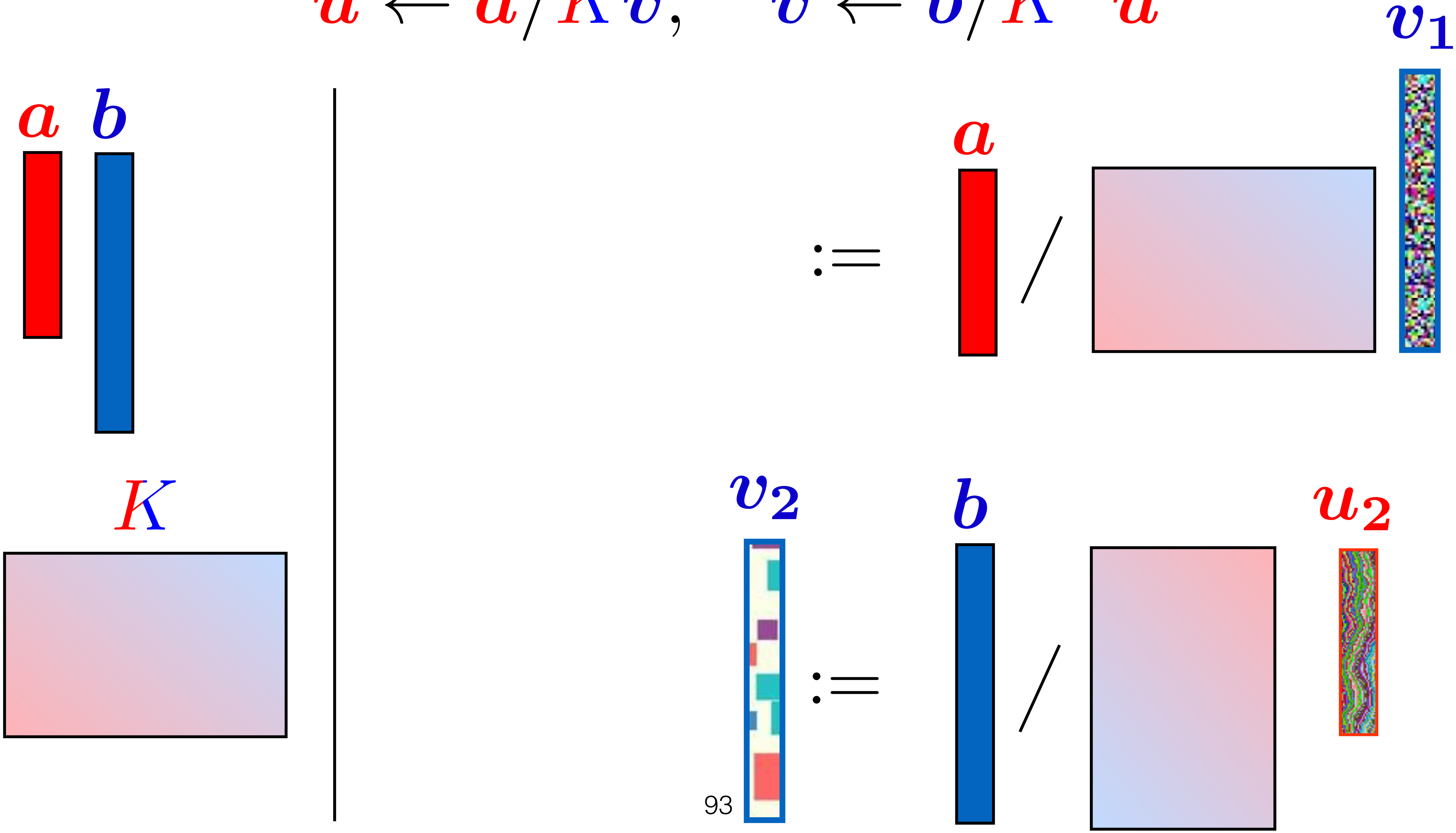
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

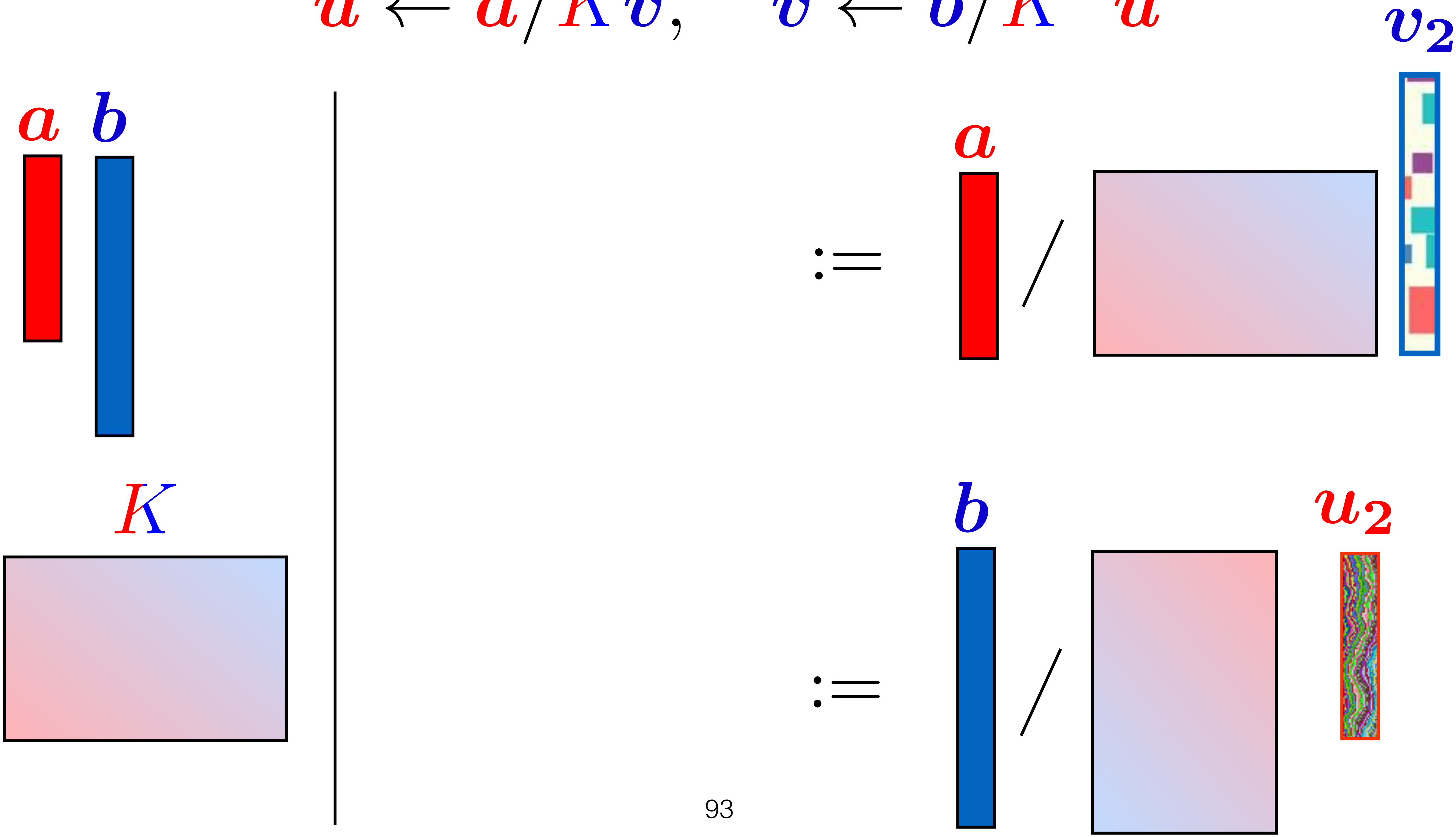
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

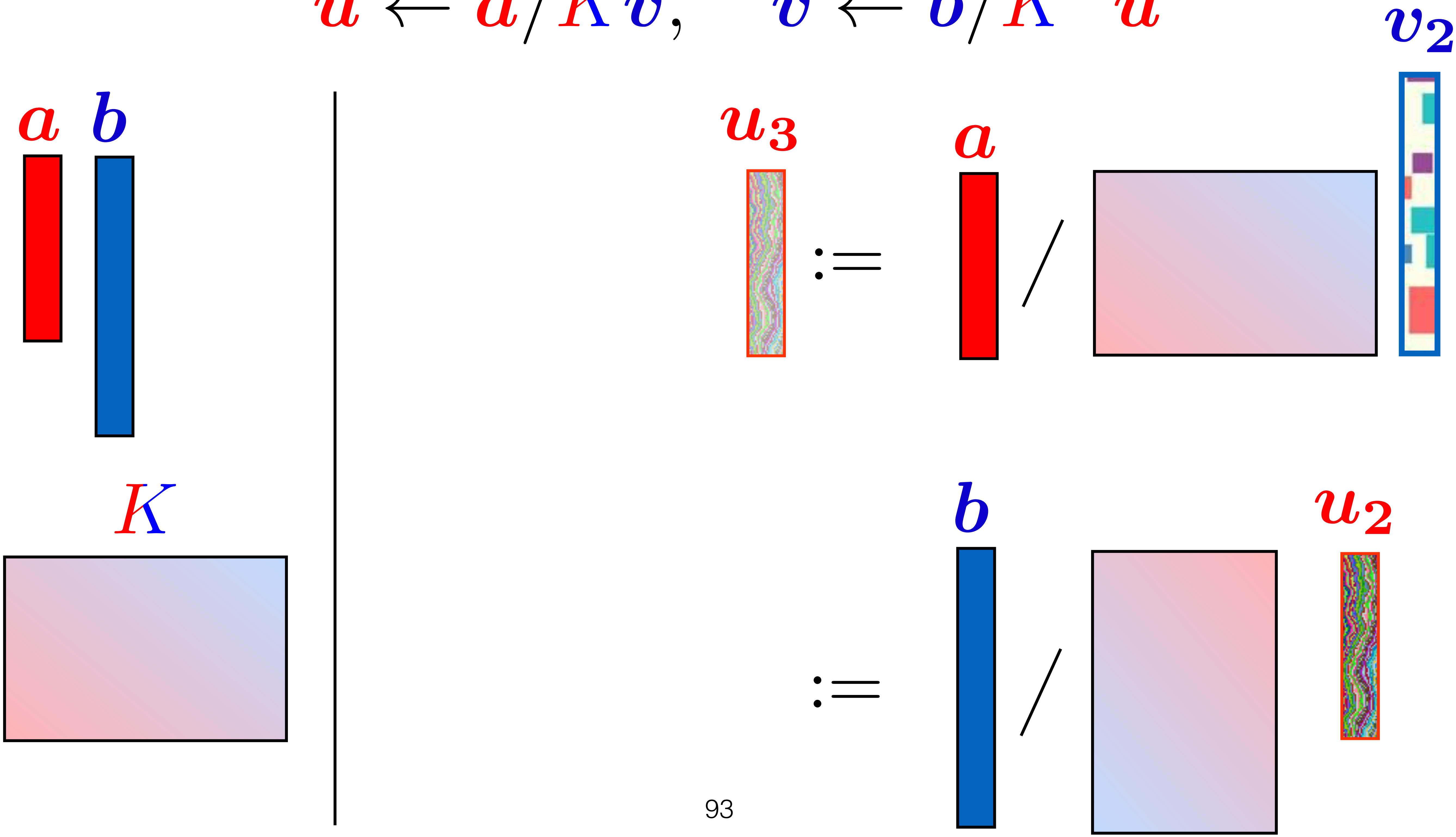
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

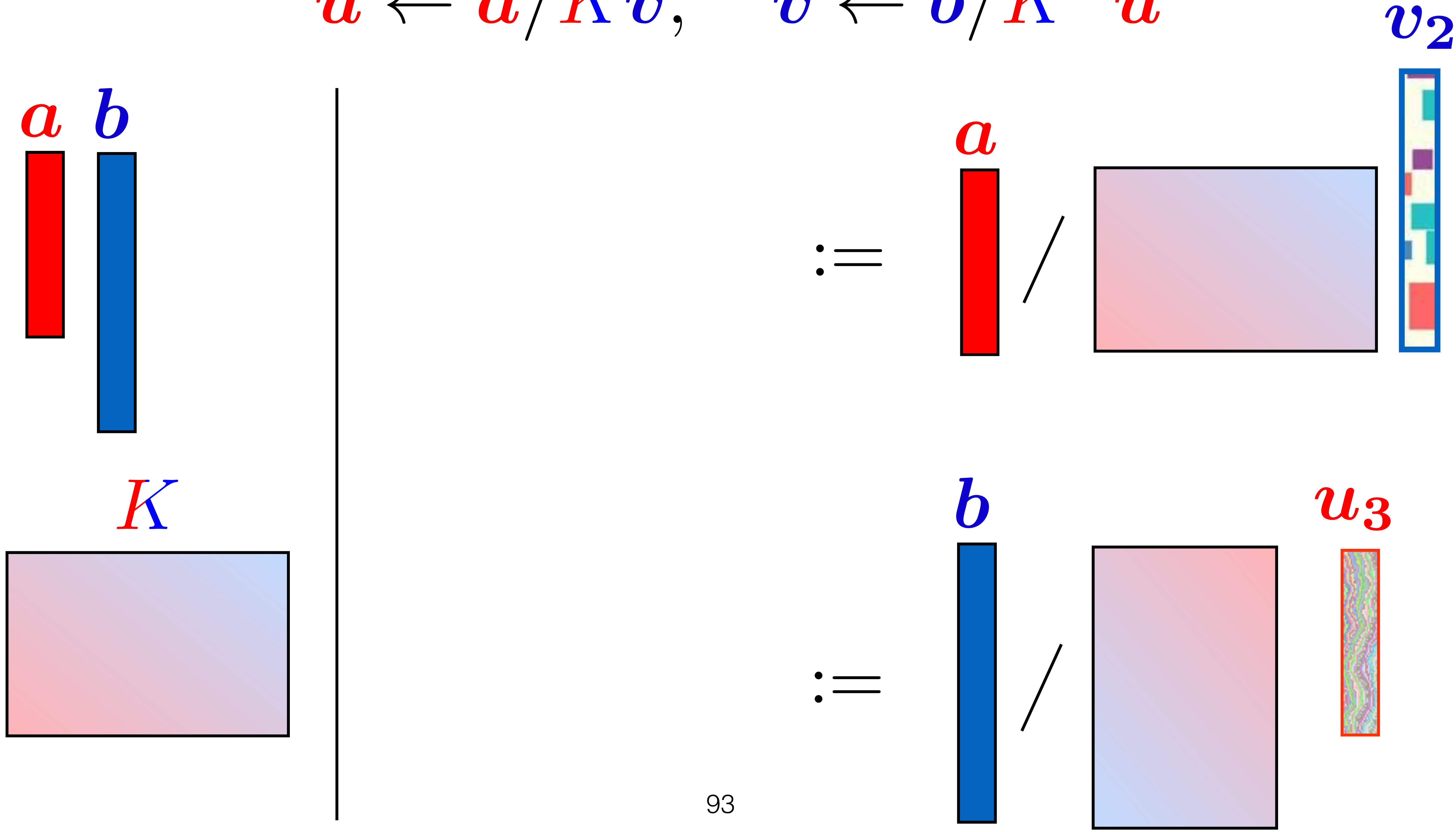
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

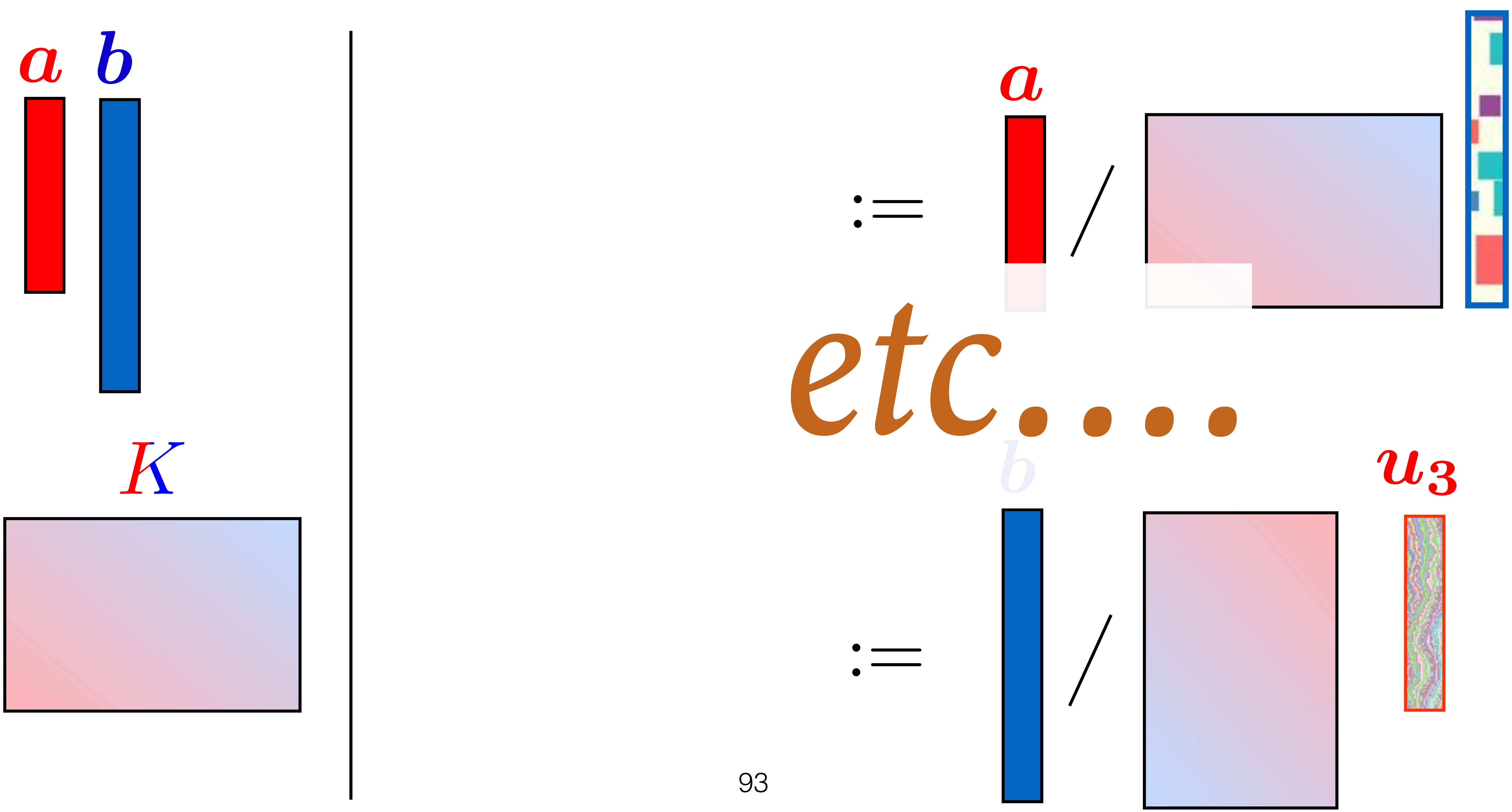
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

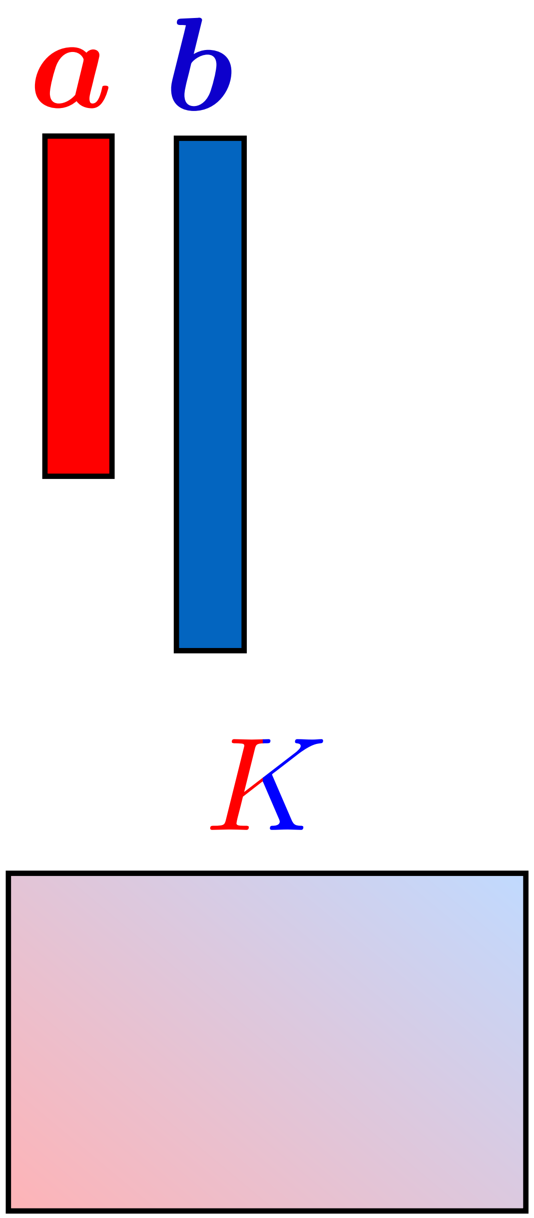
- **[Sinkhorn'64]** fixed-point iterations for $(\boldsymbol{u}, \boldsymbol{v})$

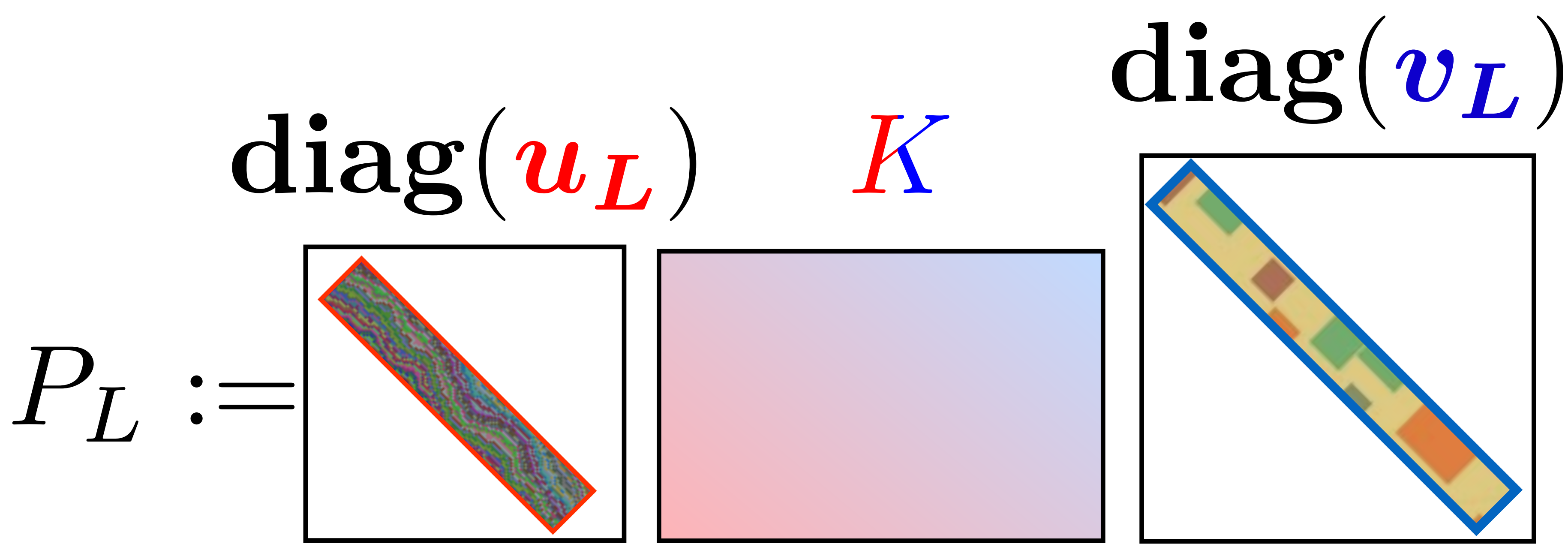
$$\boldsymbol{u} \leftarrow \boldsymbol{a}/K\boldsymbol{v}, \quad \boldsymbol{v} \leftarrow \boldsymbol{b}/K^T\boldsymbol{u}$$



Fast & Scalable Algorithm

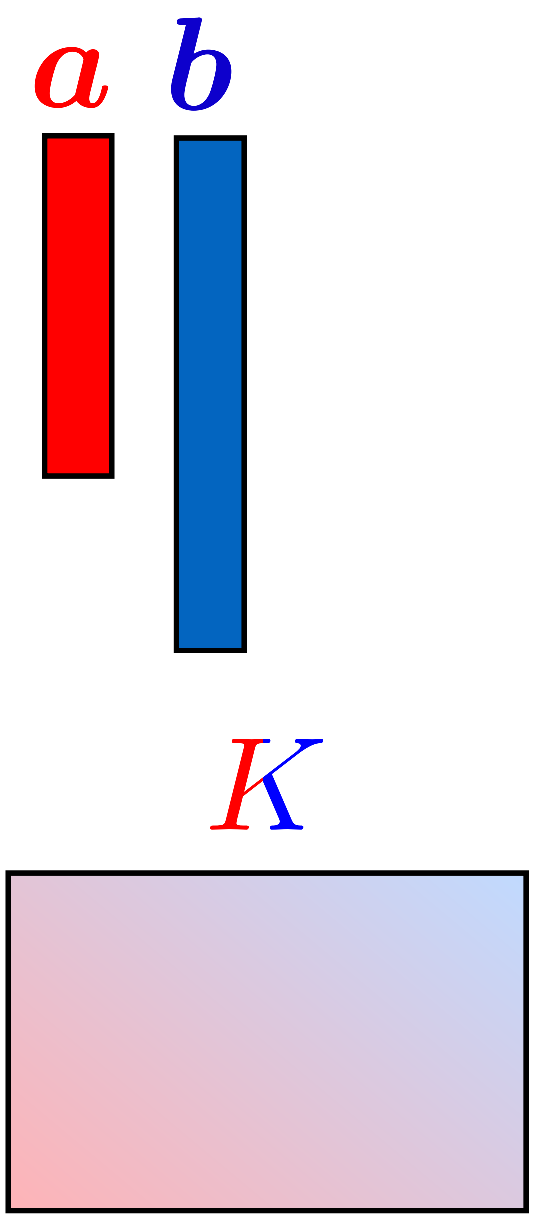
- **[Sinkhorn'64]** fixed-point iterations.





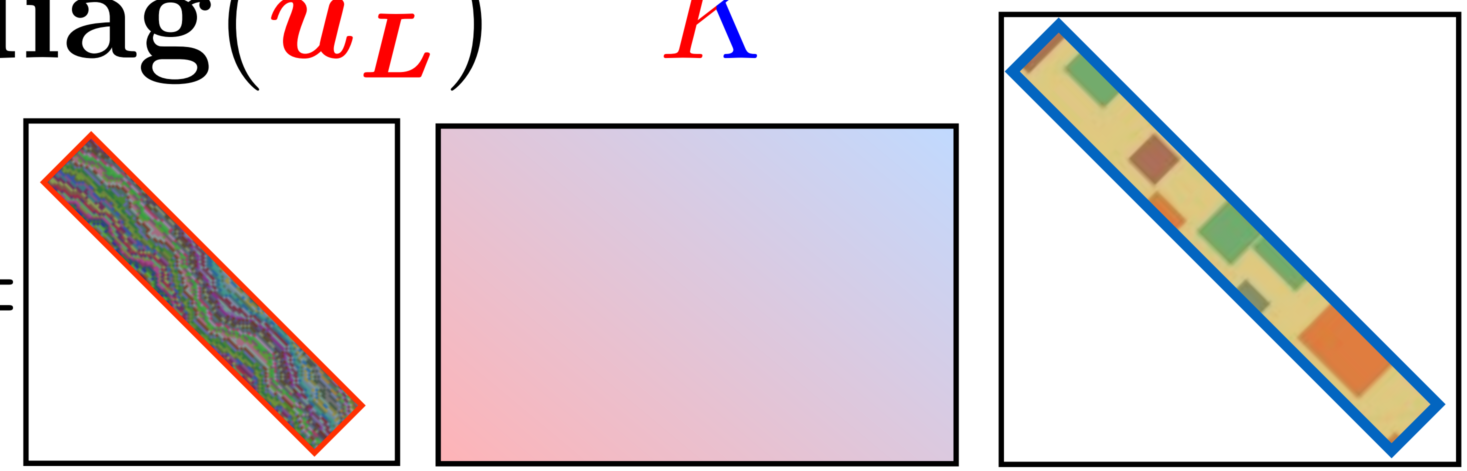
Fast & Scalable Algorithm

- **[Sinkhorn'64]** fixed-point iterations.

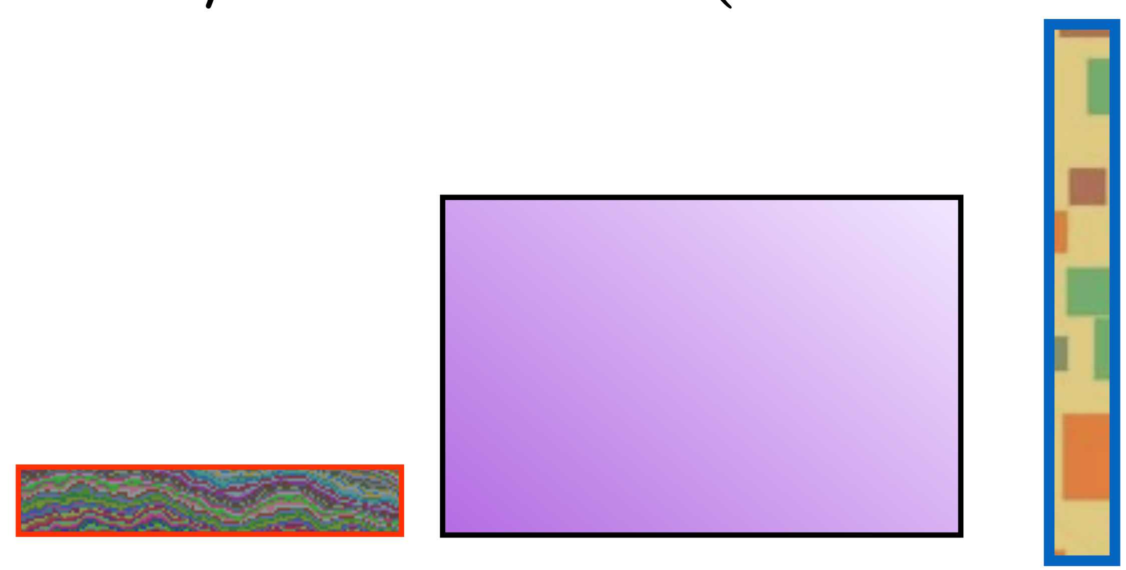


$$\mathbf{diag}(\boldsymbol{u_L}) \quad K \quad \mathbf{diag}(\boldsymbol{v_L})$$

$$P_L :=$$

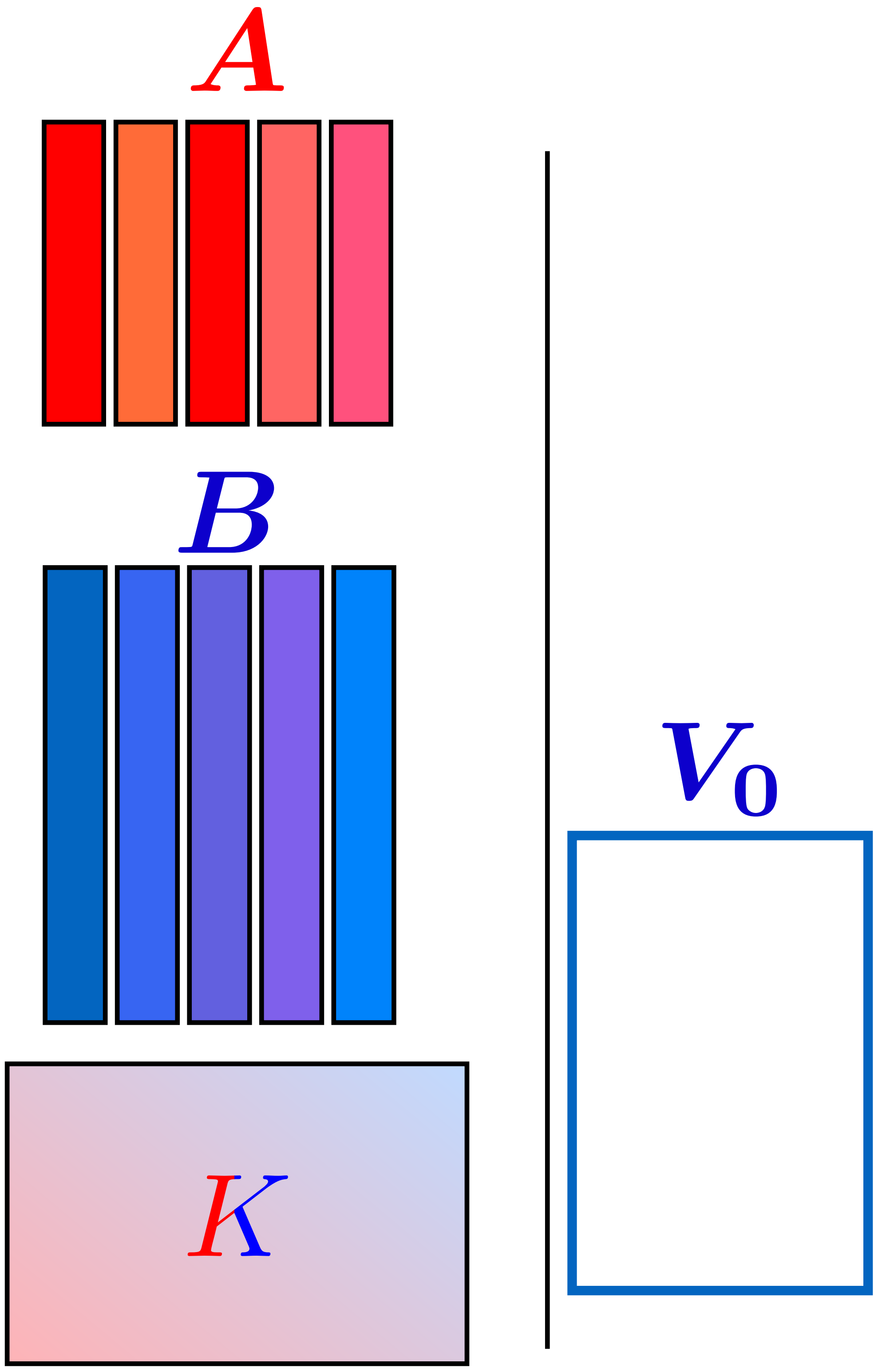


$$\langle P_L, M_{\boldsymbol{XY}} \rangle = \boldsymbol{u_L}^T (K \odot M_{\boldsymbol{XY}}) \boldsymbol{v_L}$$



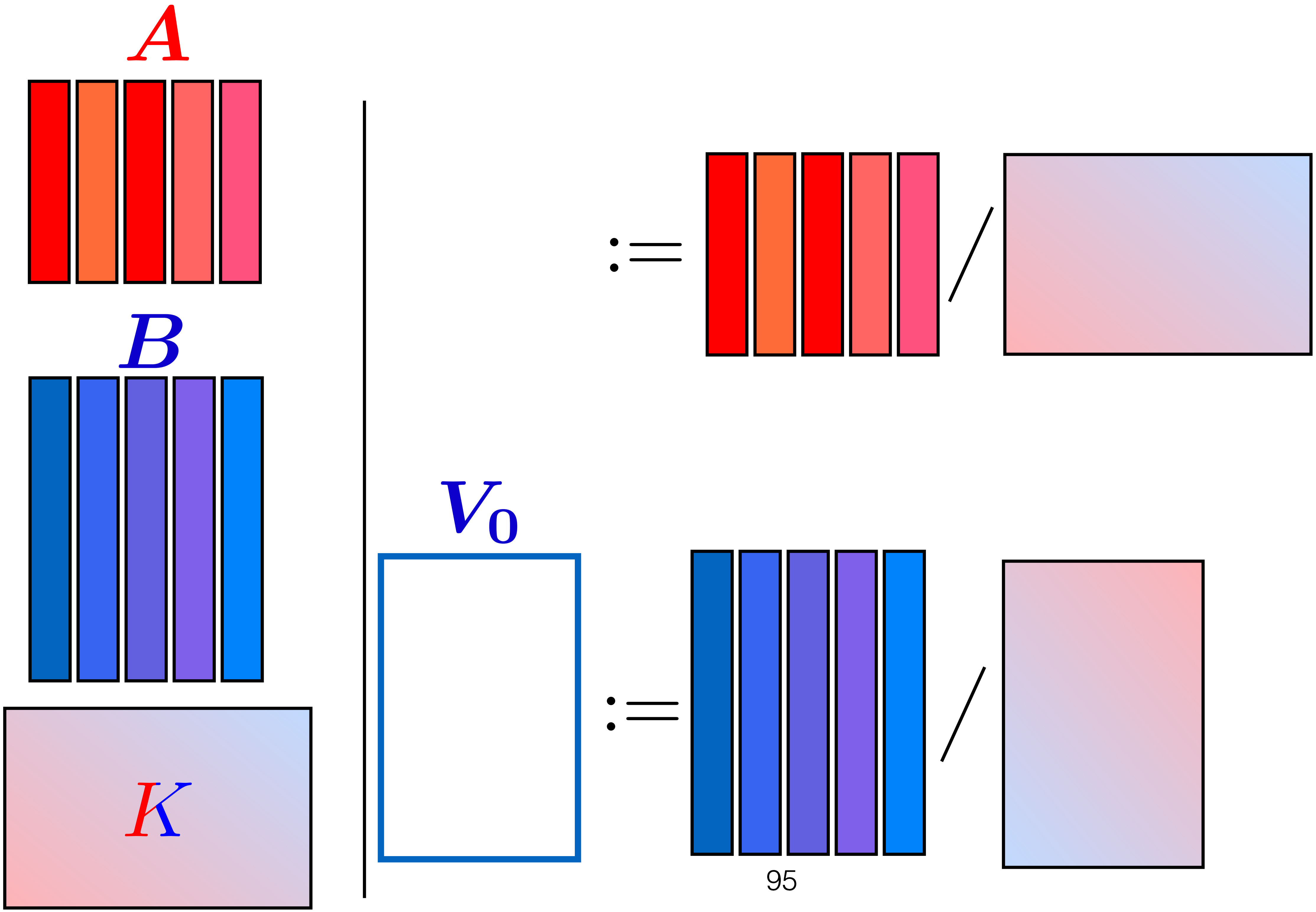
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



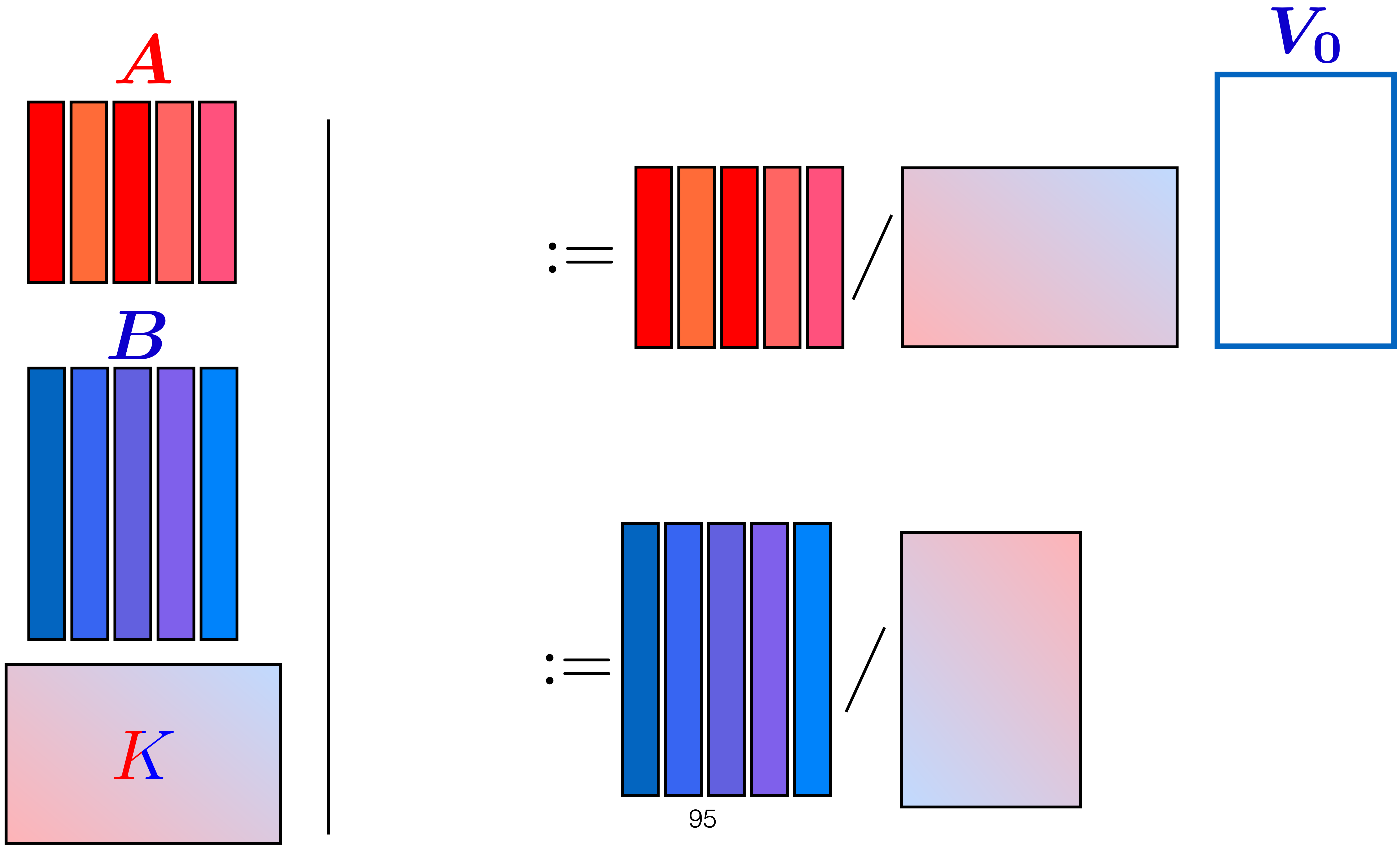
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



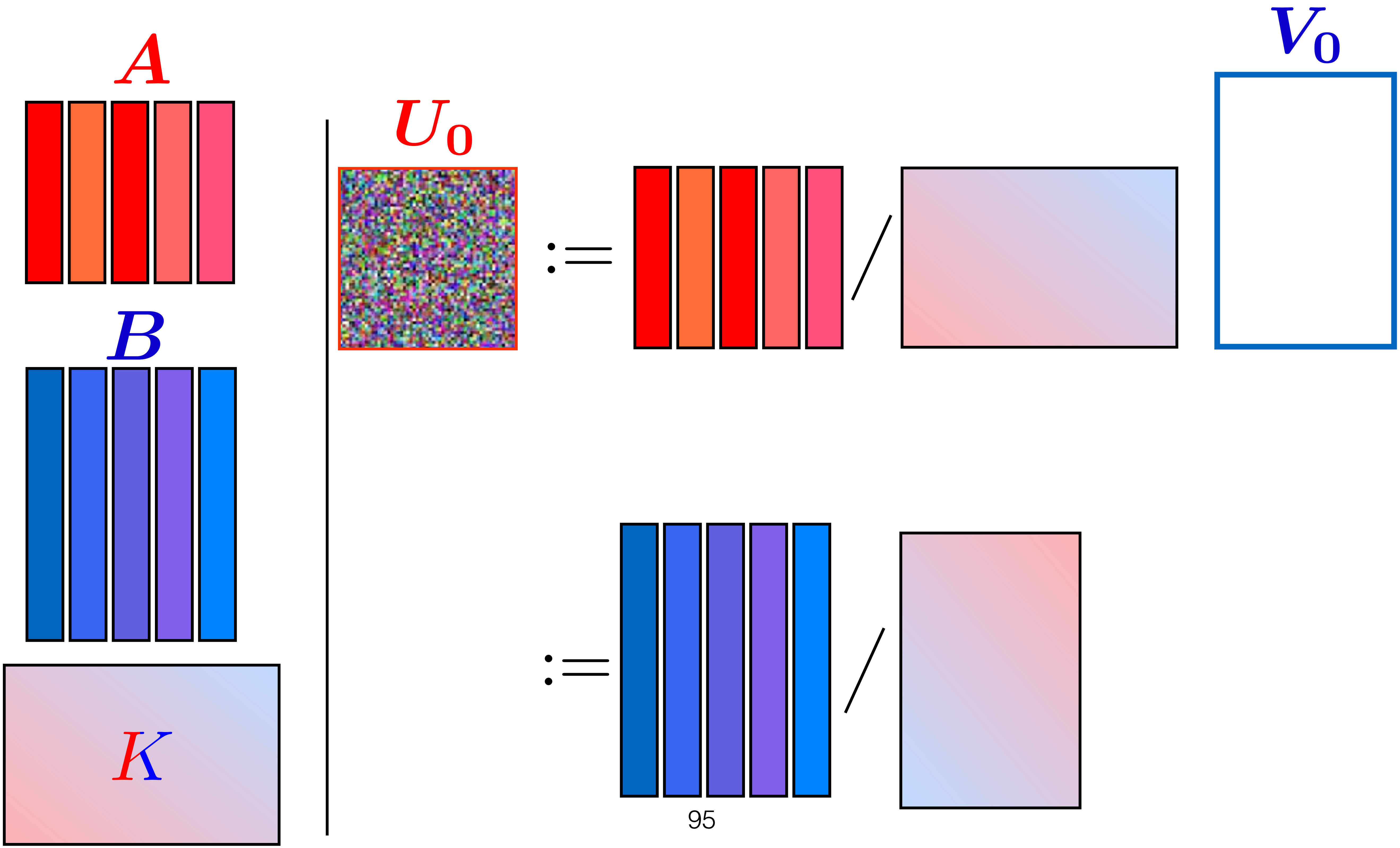
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



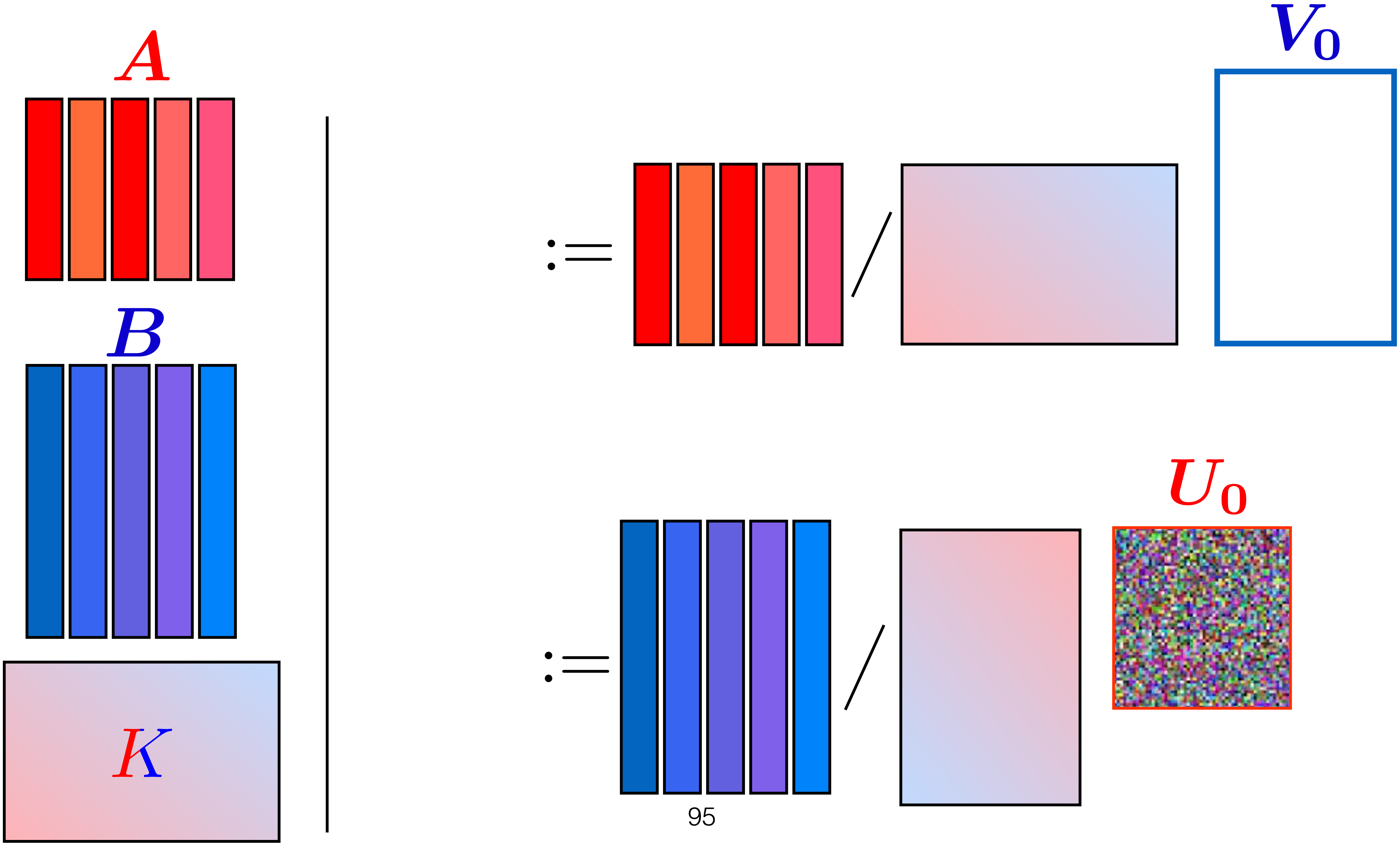
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



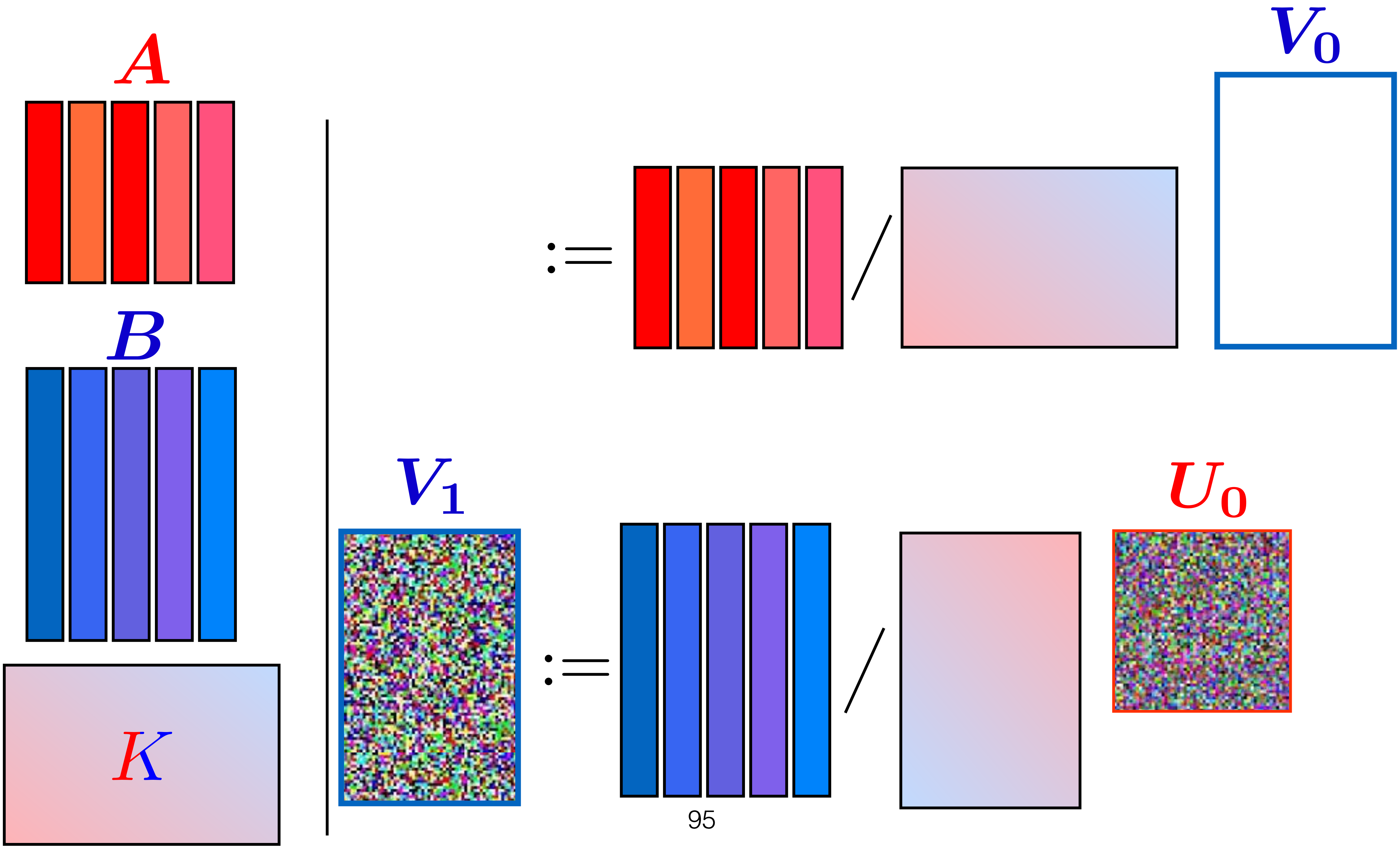
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



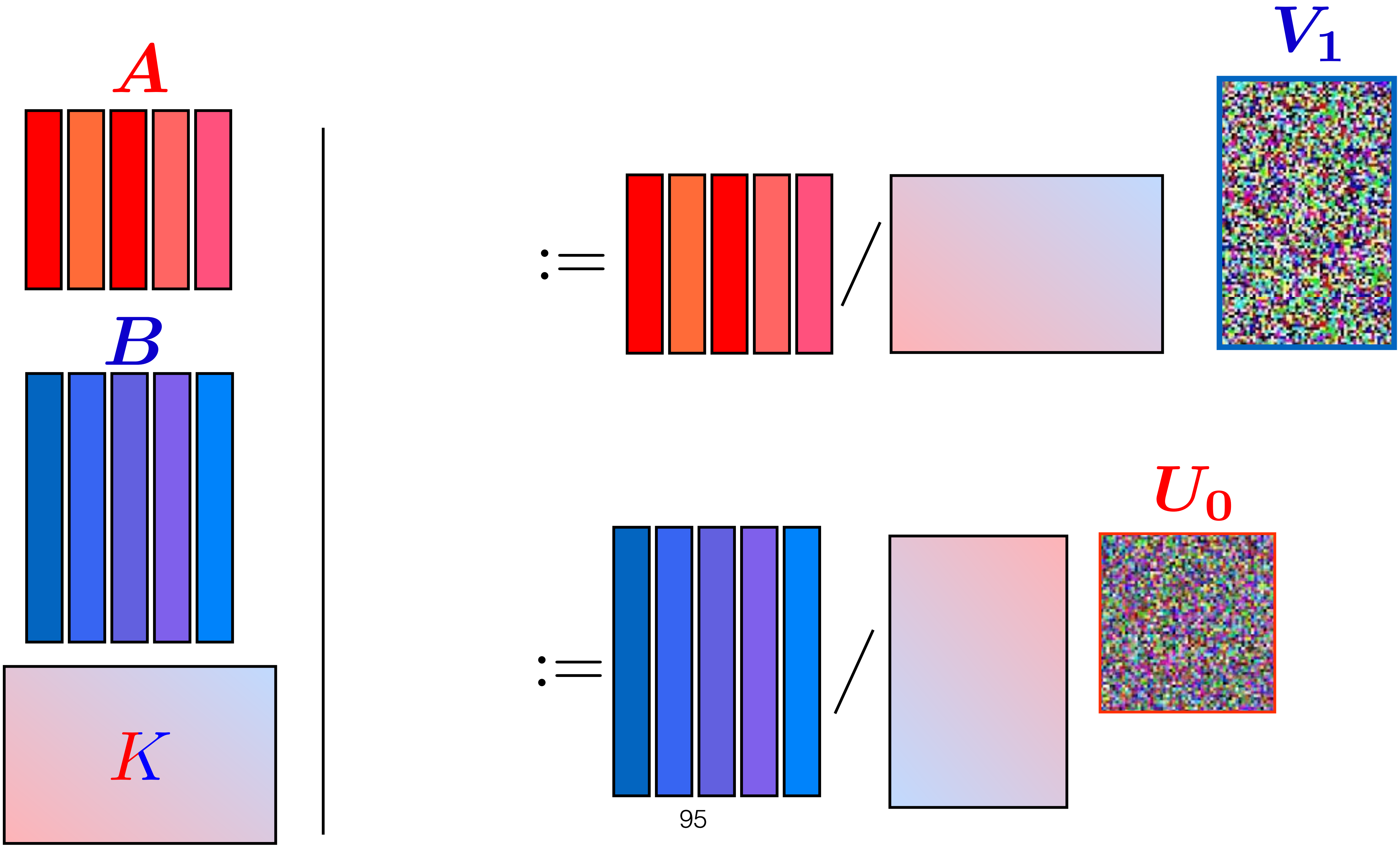
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



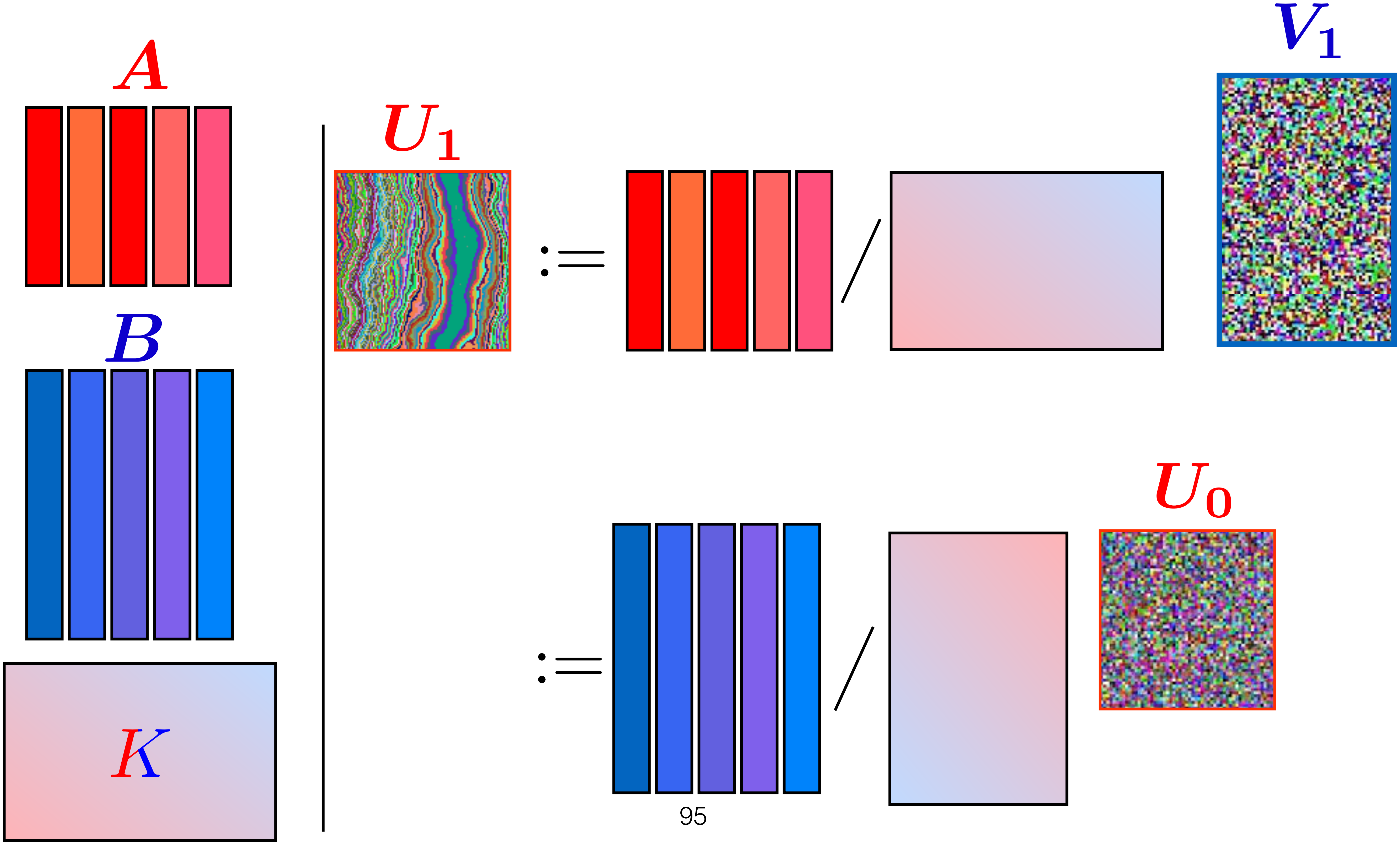
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



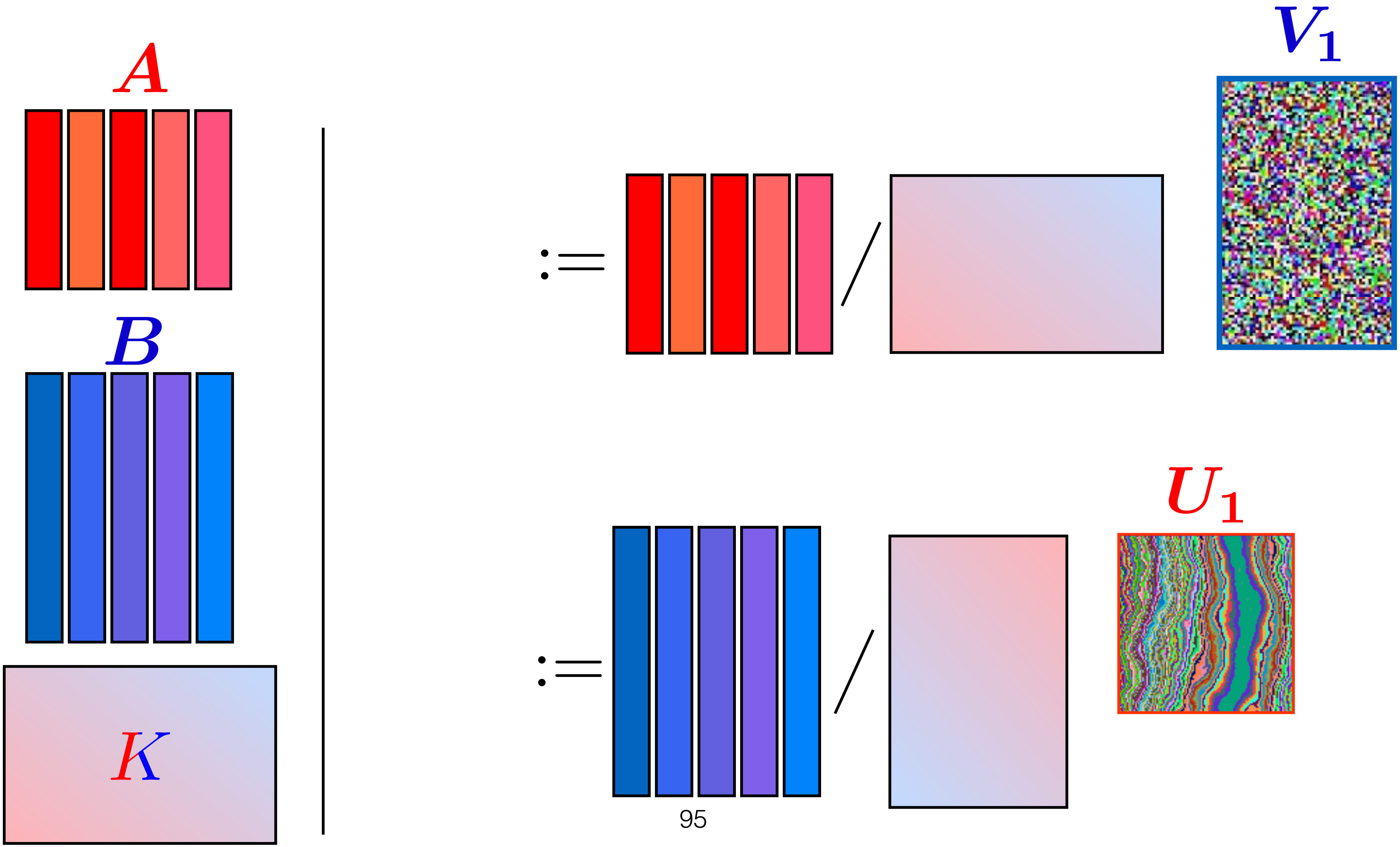
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



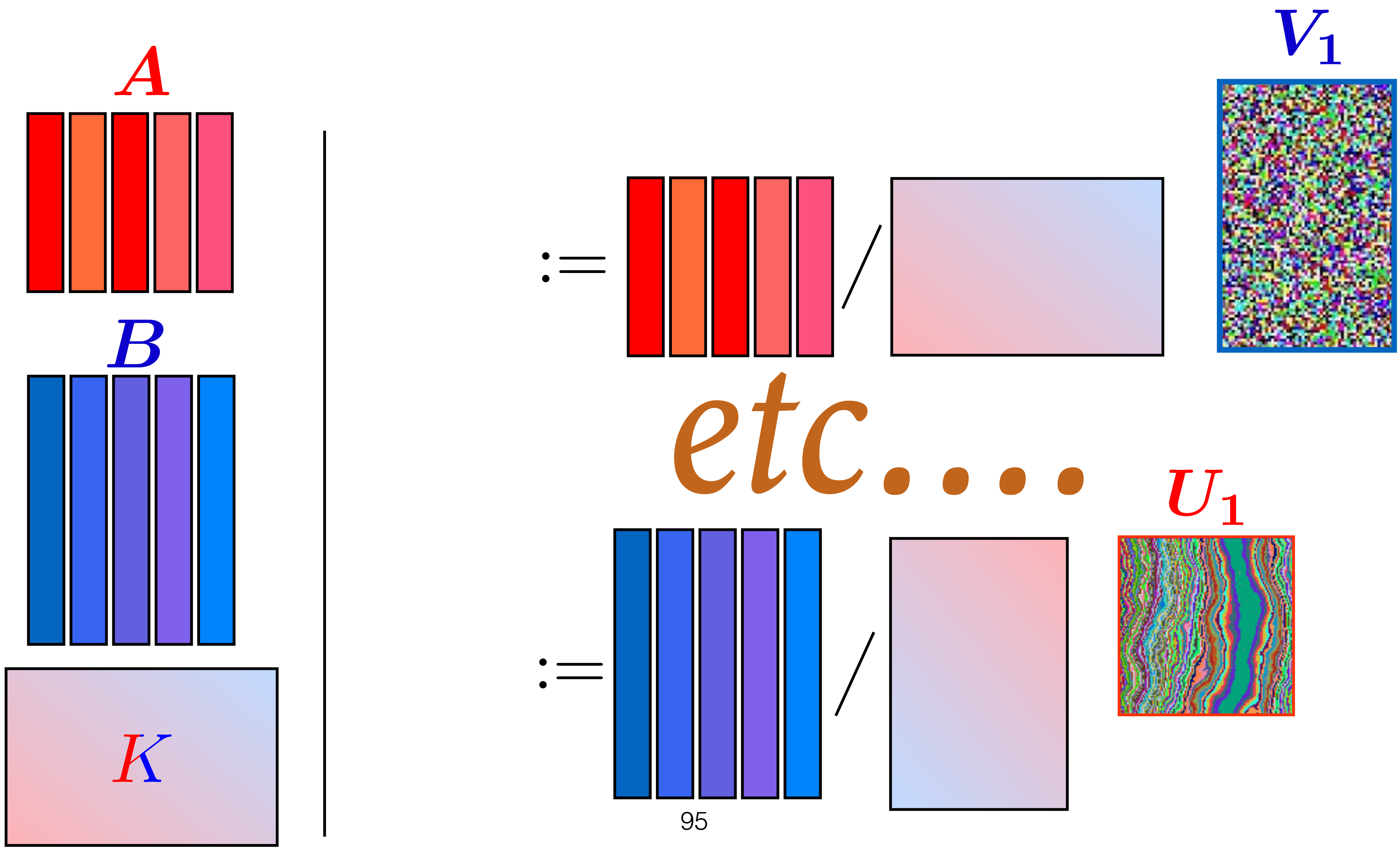
Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations

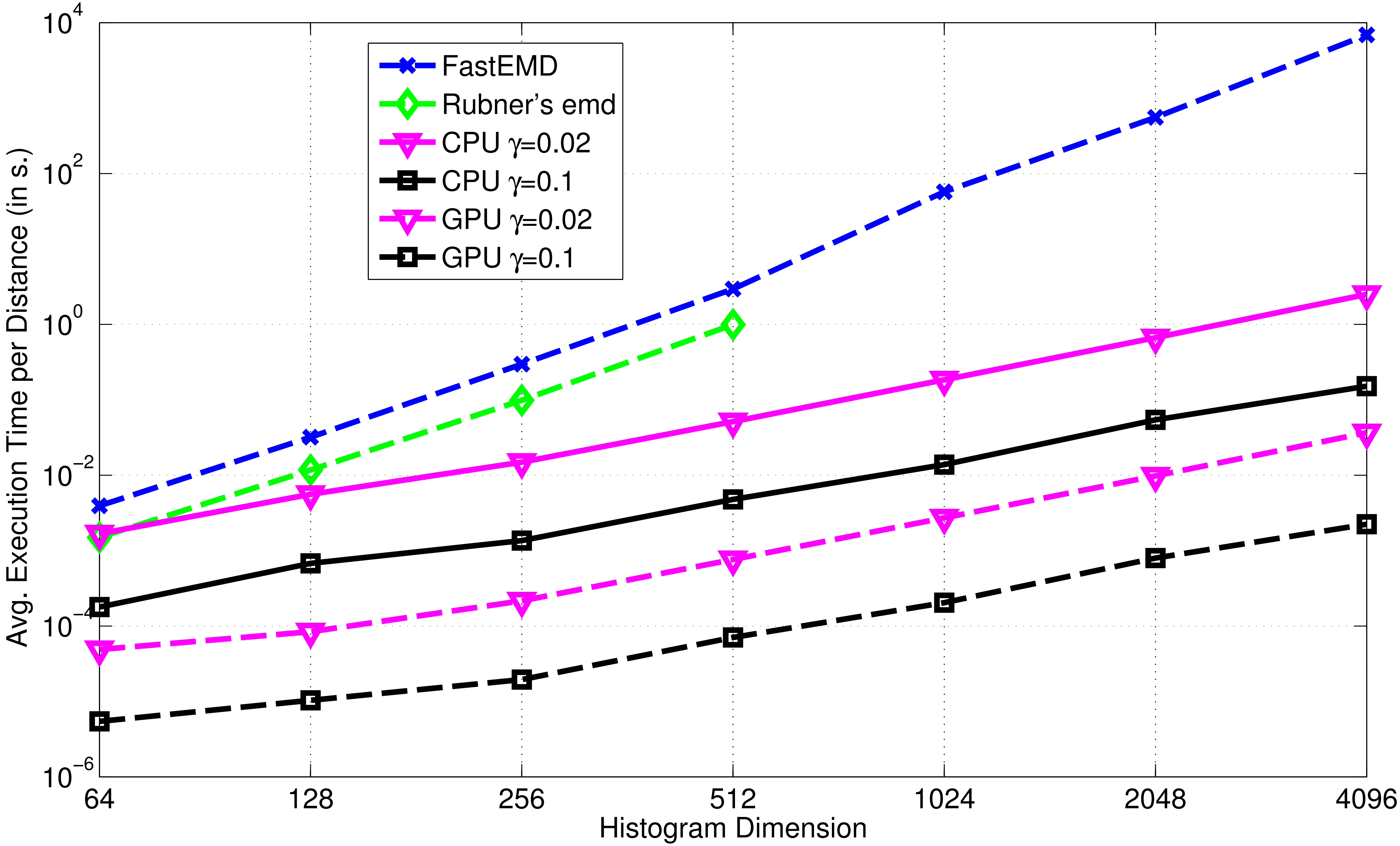


Also embarrassingly parallel

- **[Sinkhorn'64]** with *matrix* fixed-point iterations



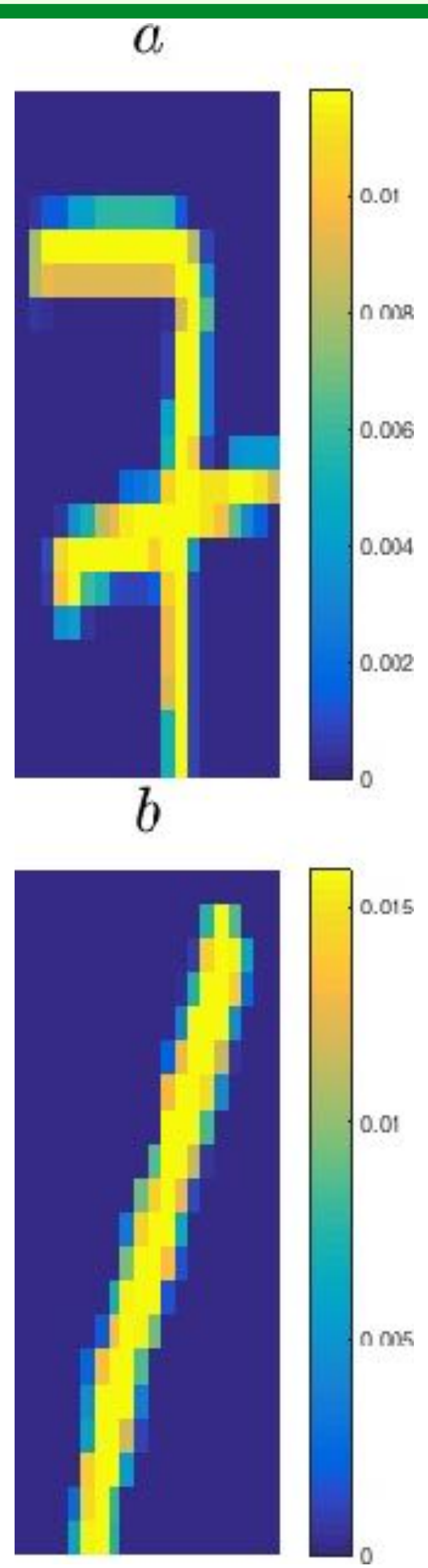
Very Fast EMD Approx. Solver



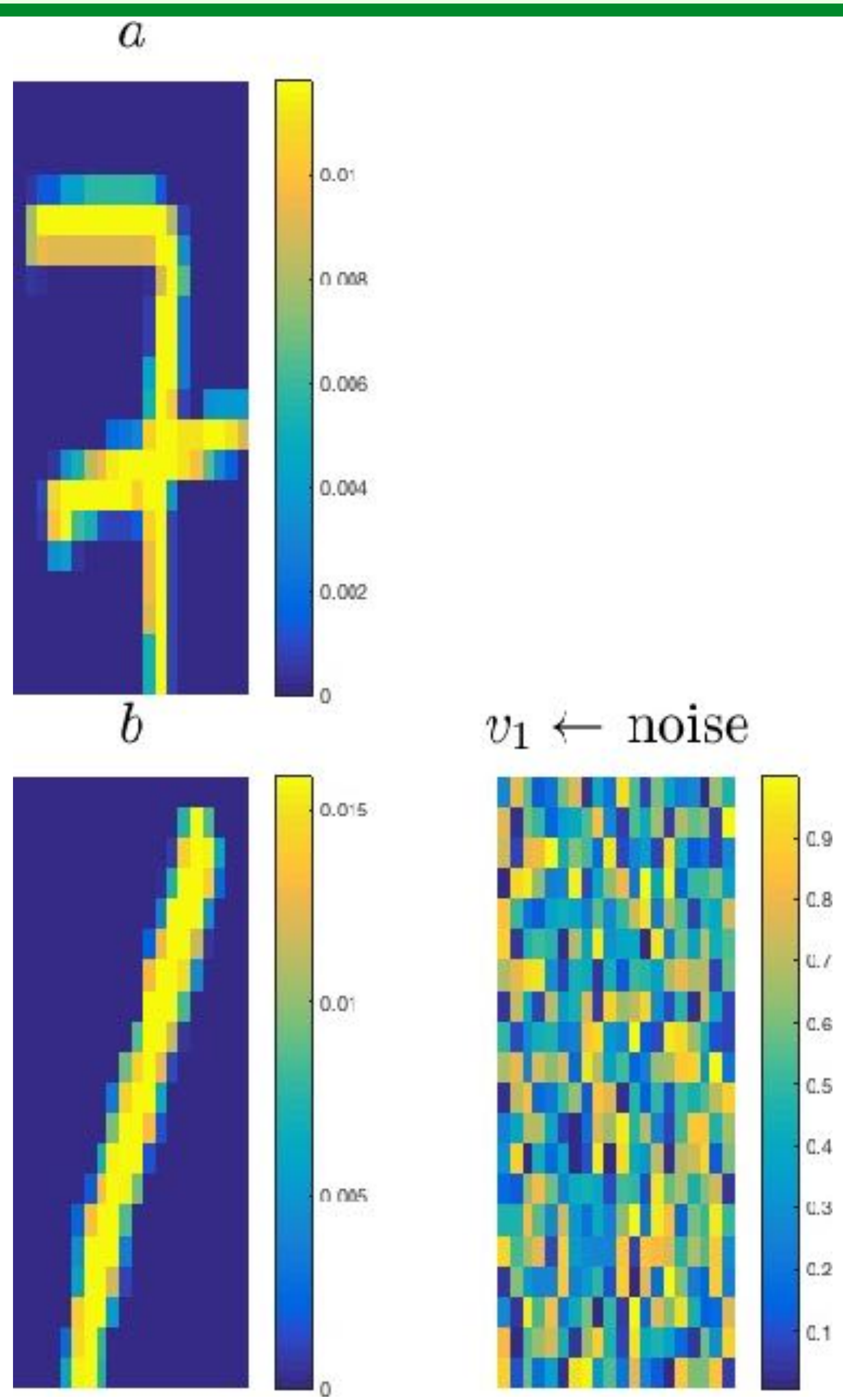
Note. $(\Omega, \boldsymbol{D})$ is a random graph with shortest path metric, histograms sampled uniformly on simplex, Sinkhorn tolerance 10^{-2}.

Very Fast EMD Approx. Solver

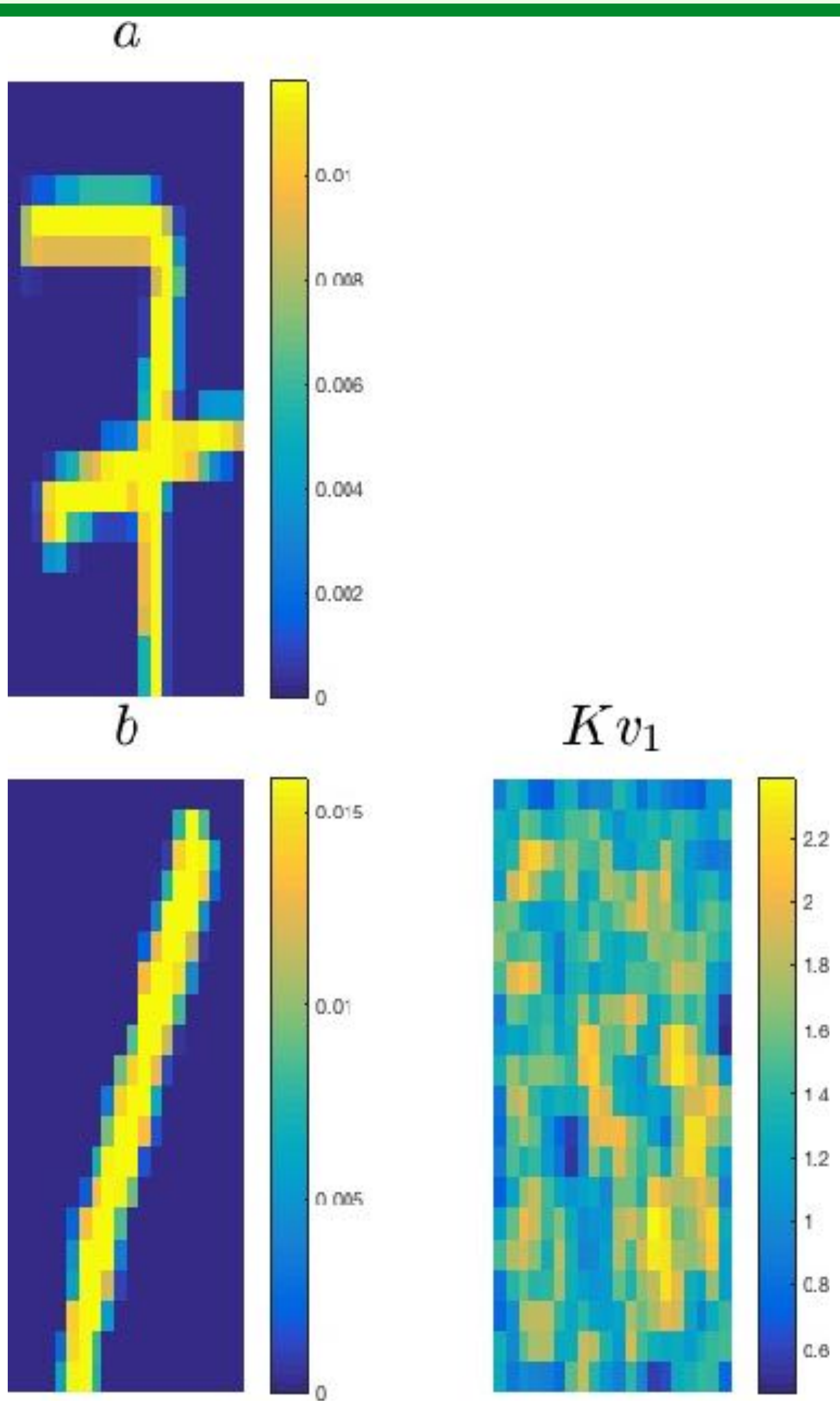
Very Fast EMD Approx. Solver



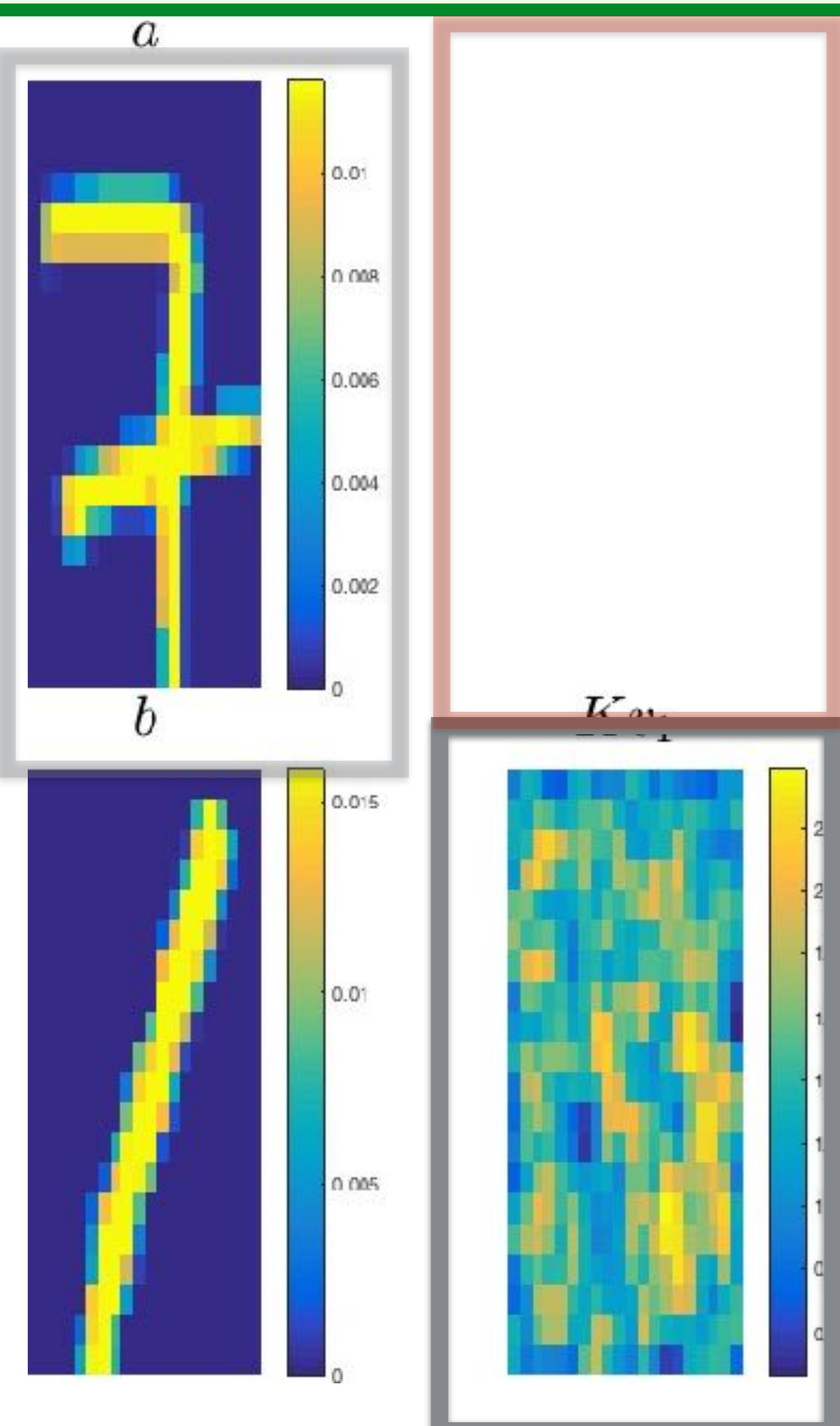
Very Fast EMD Approx. Solver



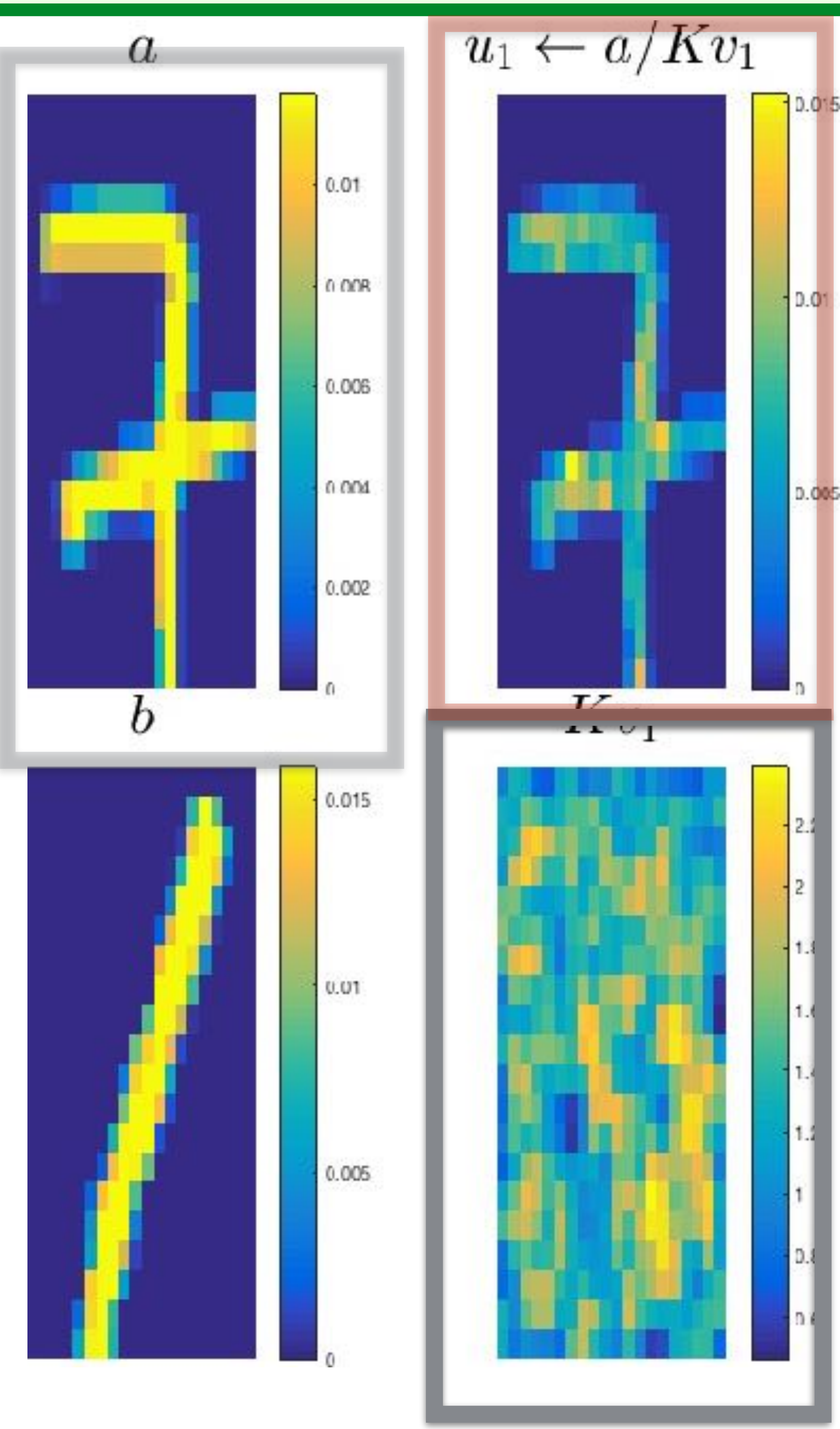
Very Fast EMD Approx. Solver



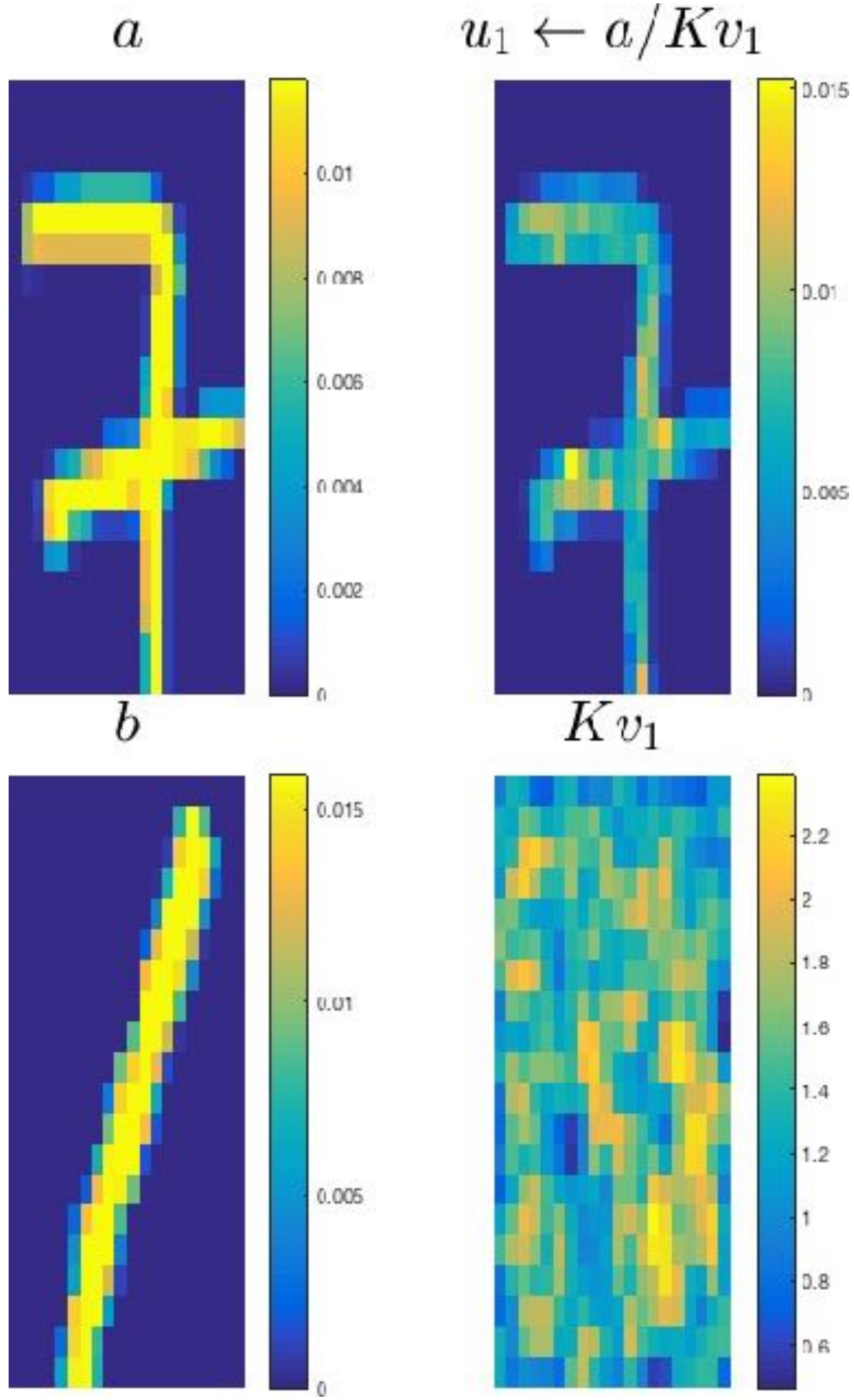
Very Fast EMD Approx. Solver



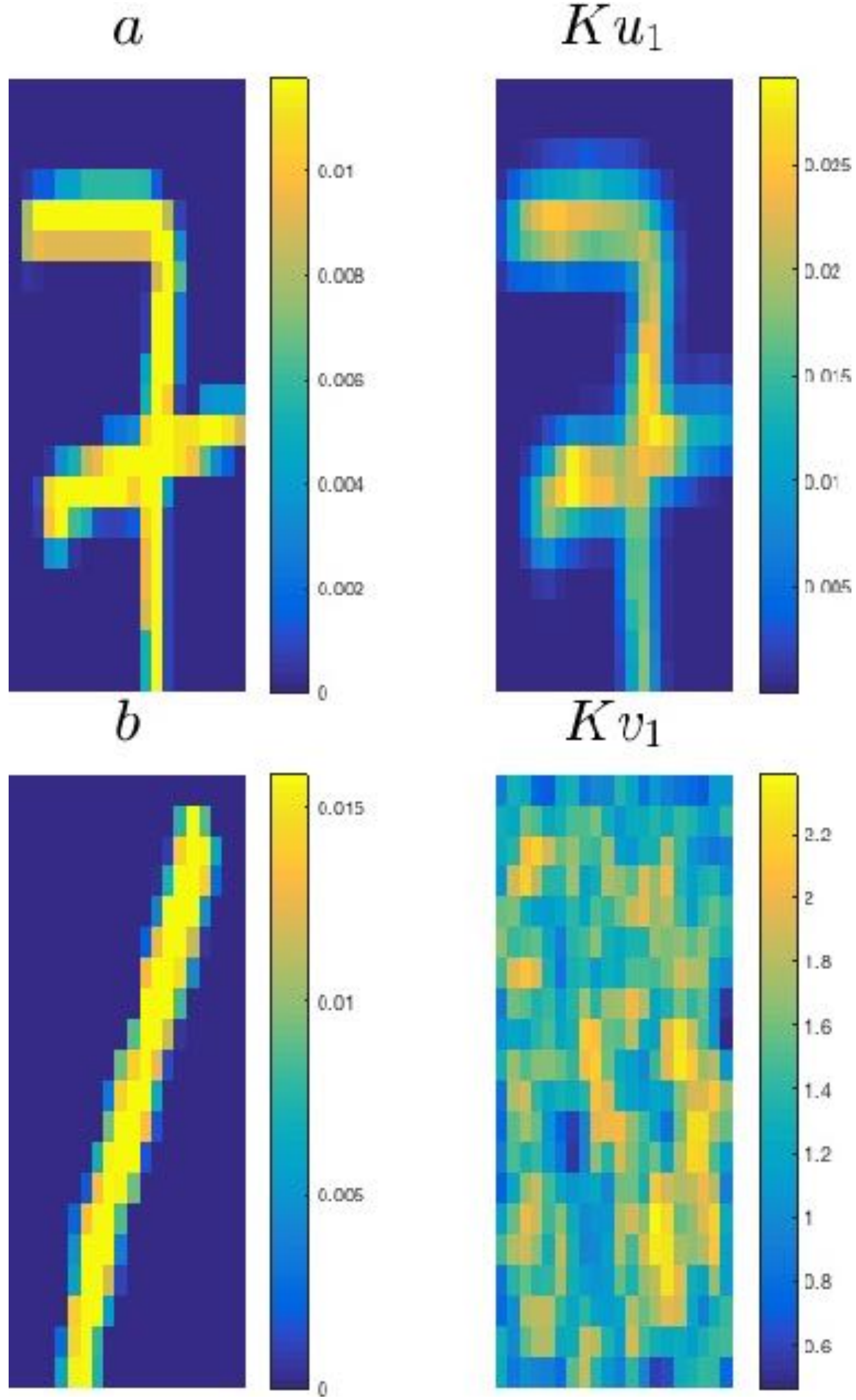
Very Fast EMD Approx. Solver



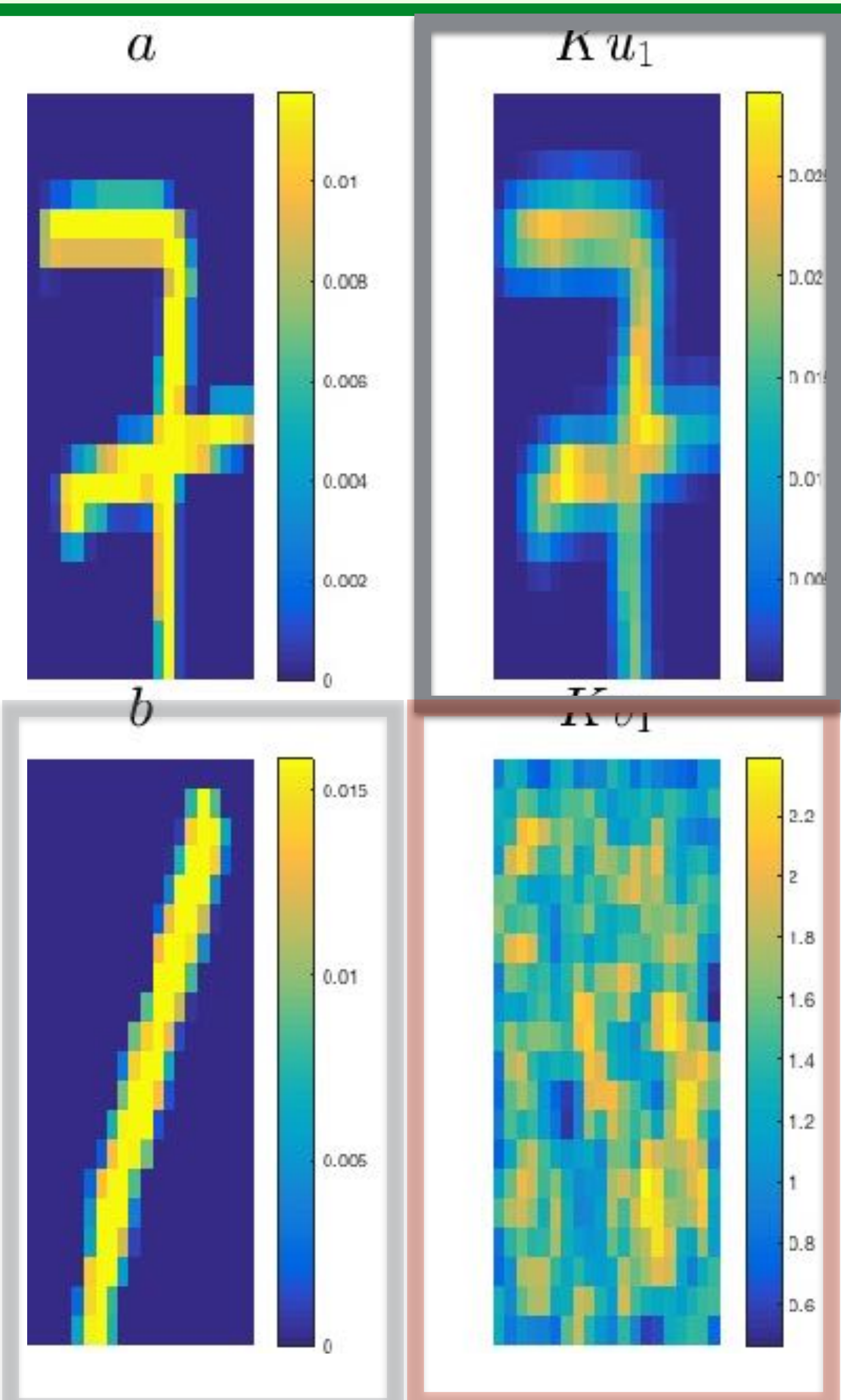
Very Fast EMD Approx. Solver



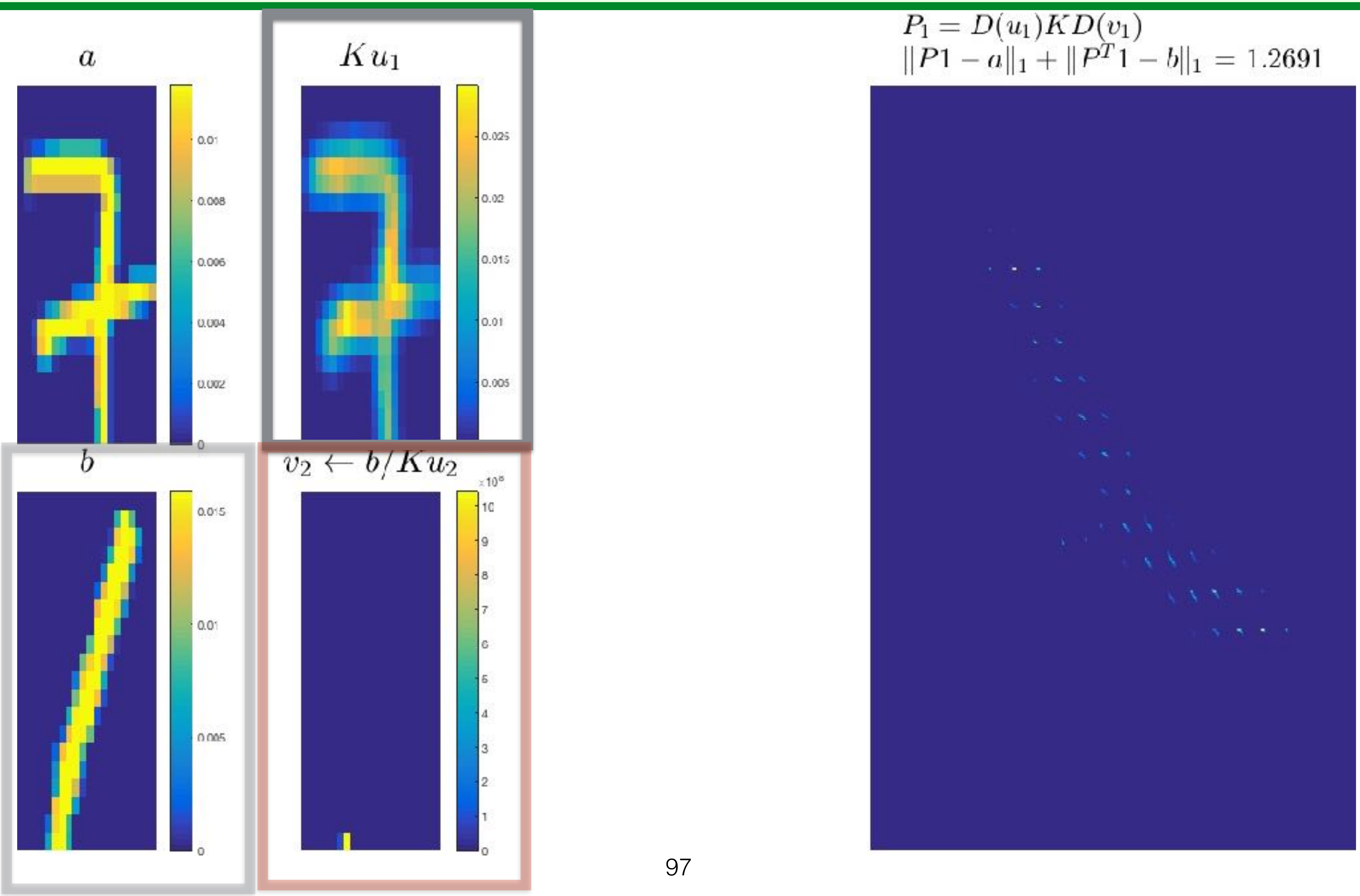
Very Fast EMD Approx. Solver



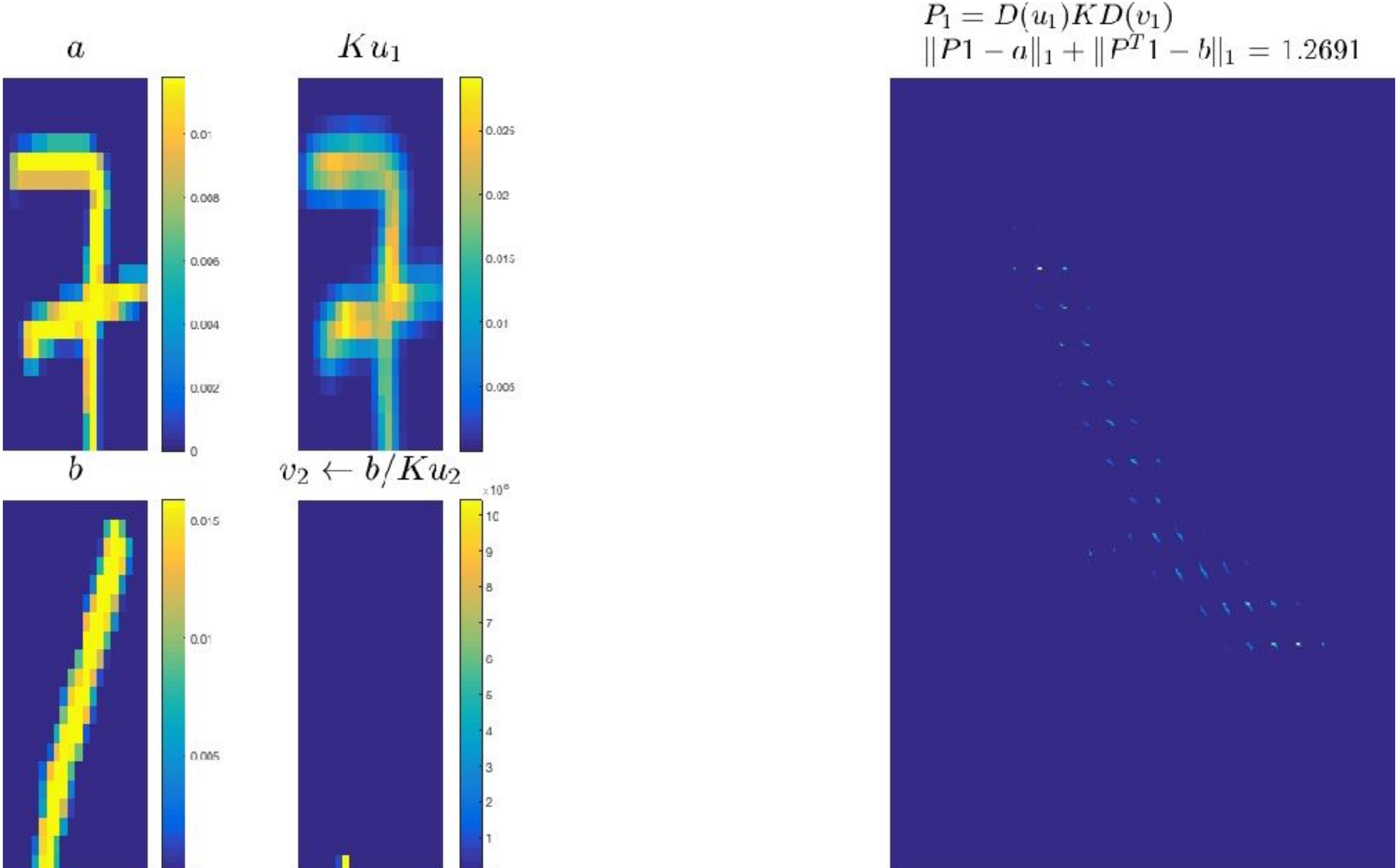
Very Fast EMD Approx. Solver



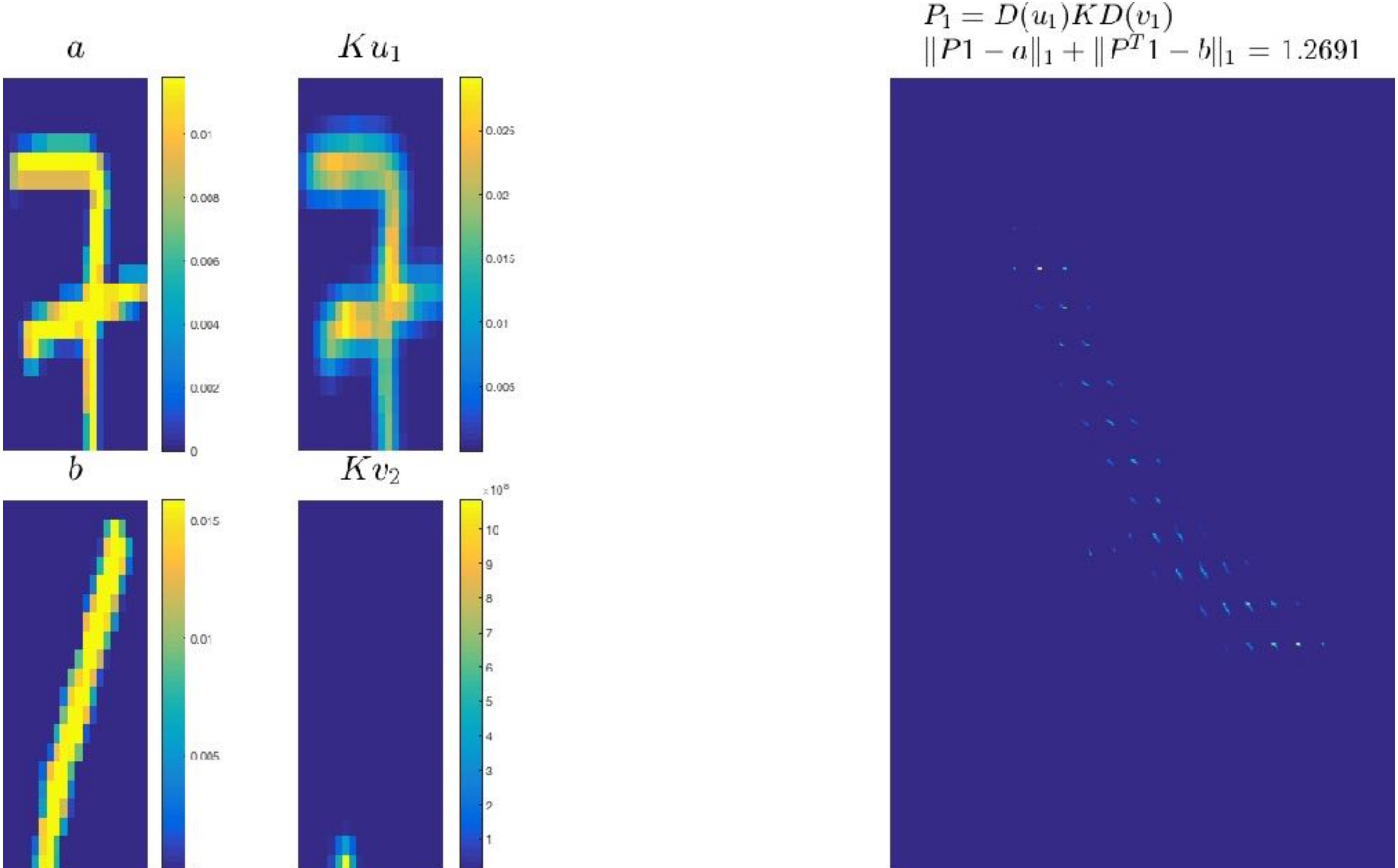
Very Fast EMD Approx. Solver



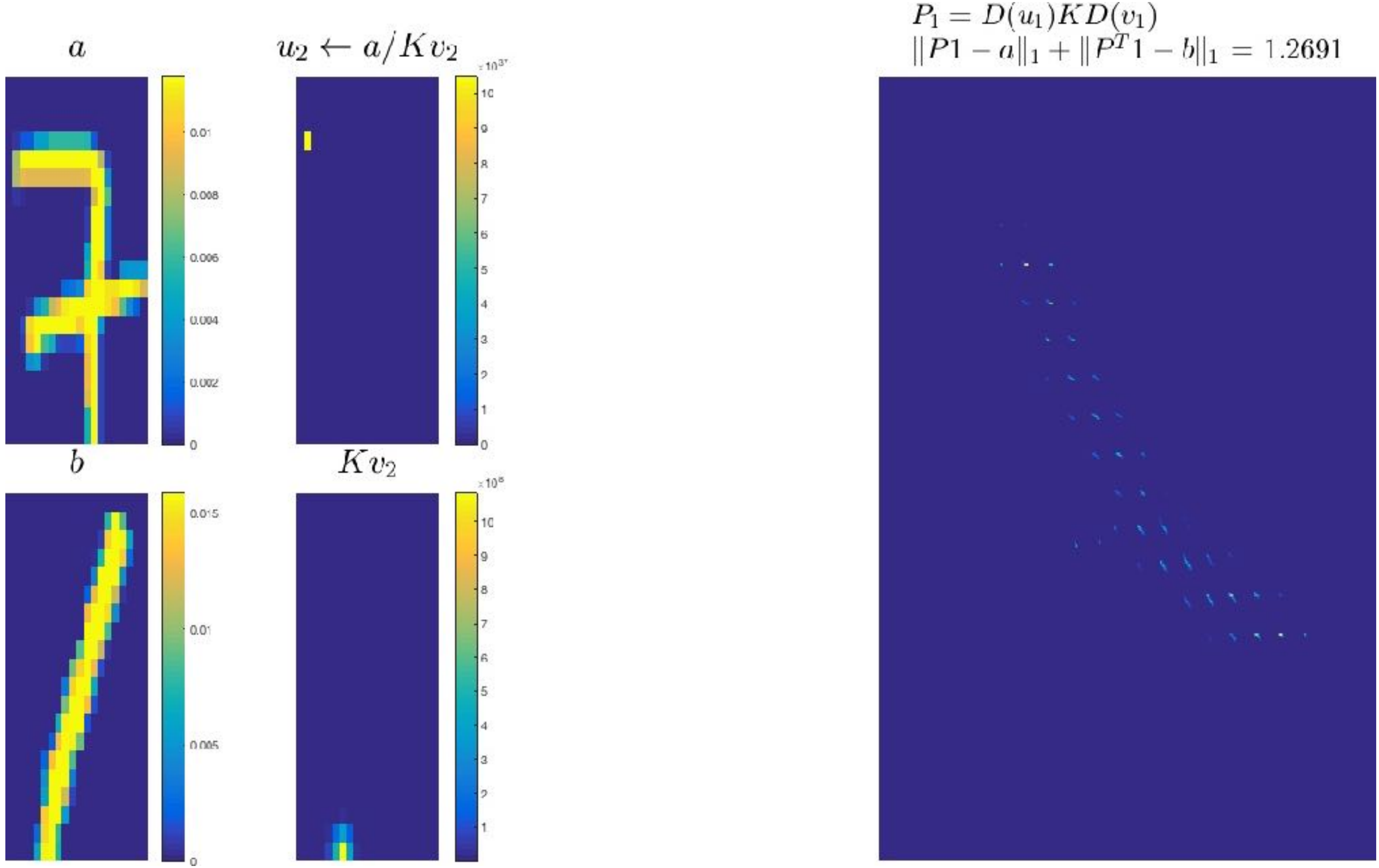
Very Fast EMD Approx. Solver



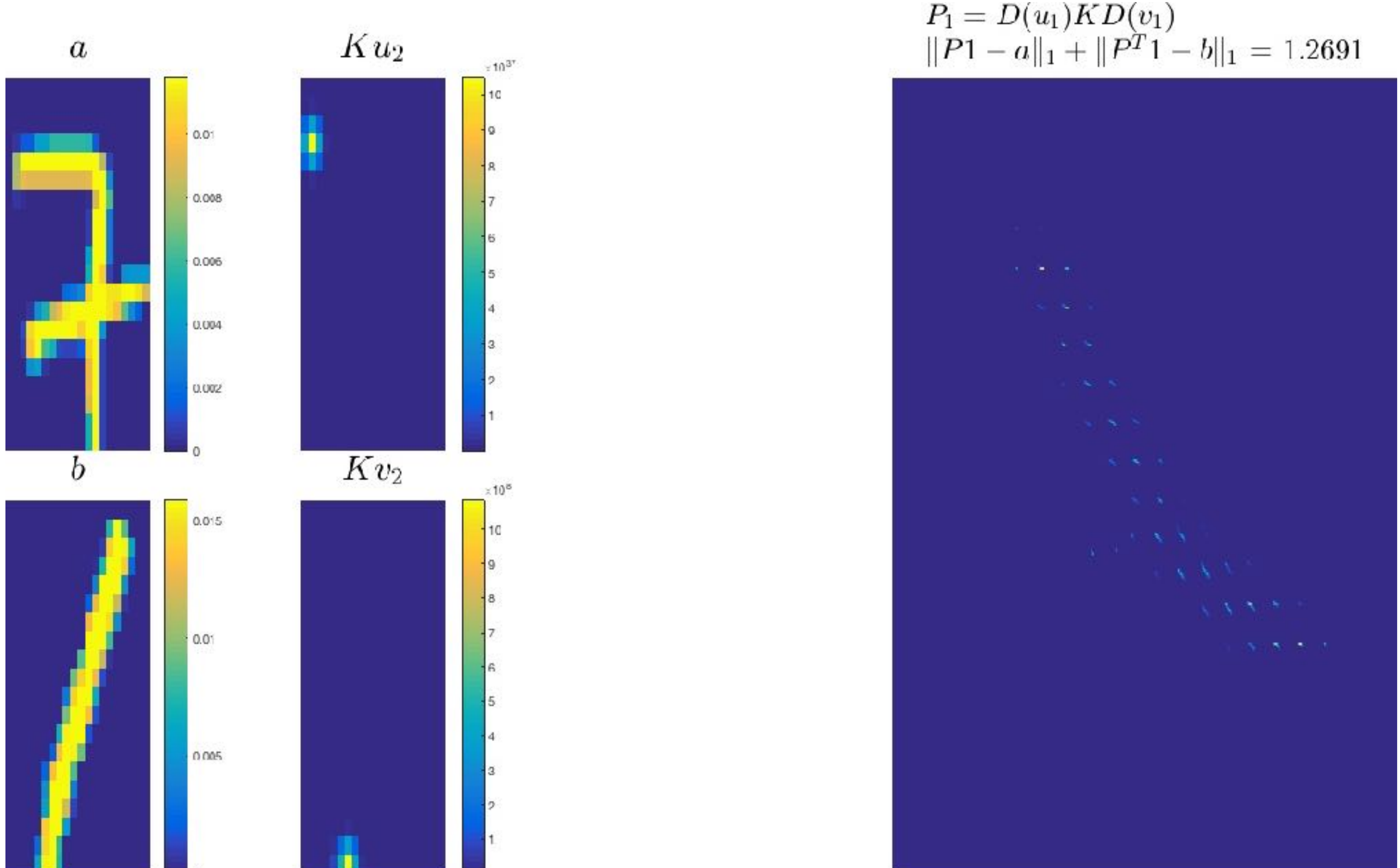
Very Fast EMD Approx. Solver



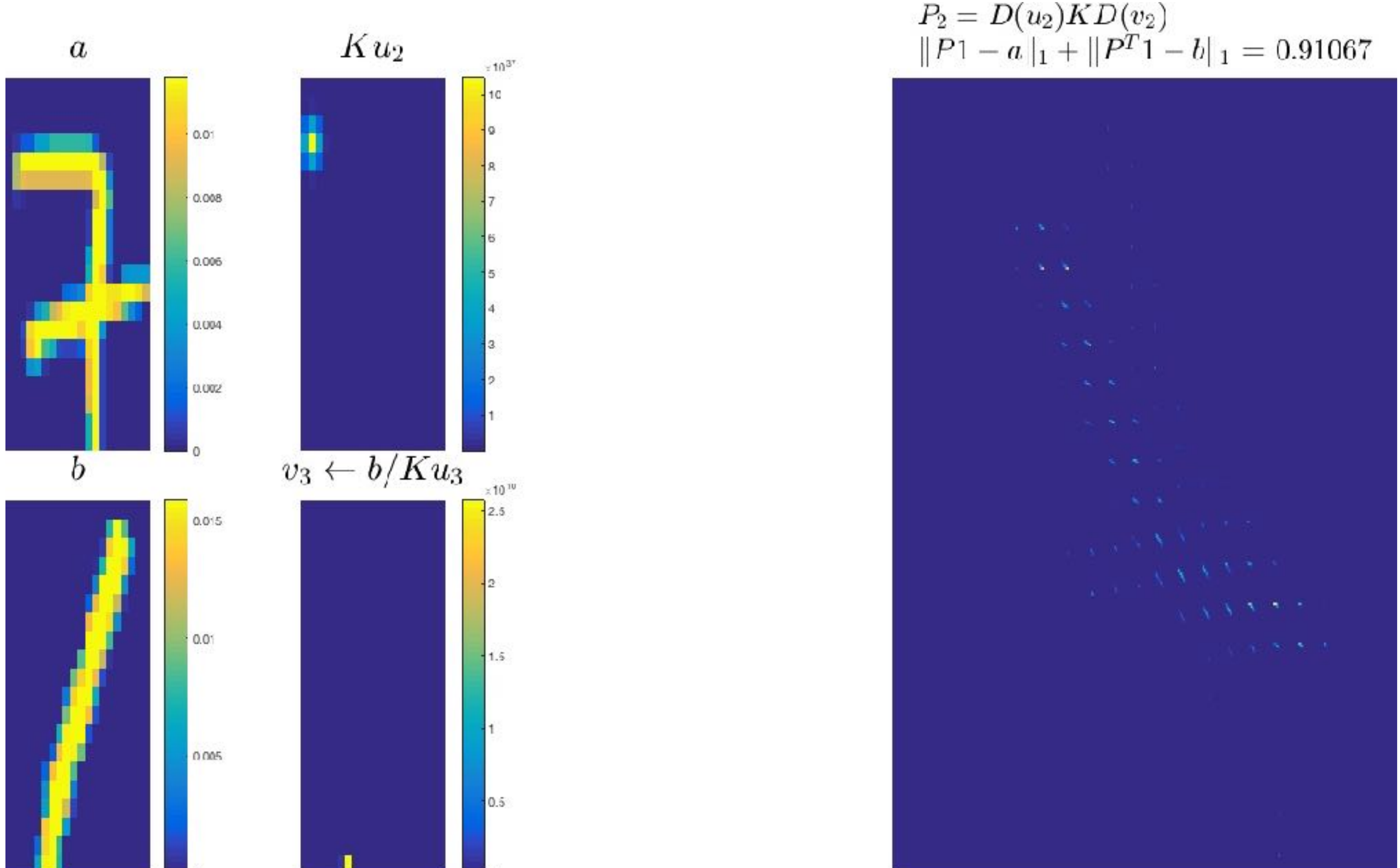
Very Fast EMD Approx. Solver



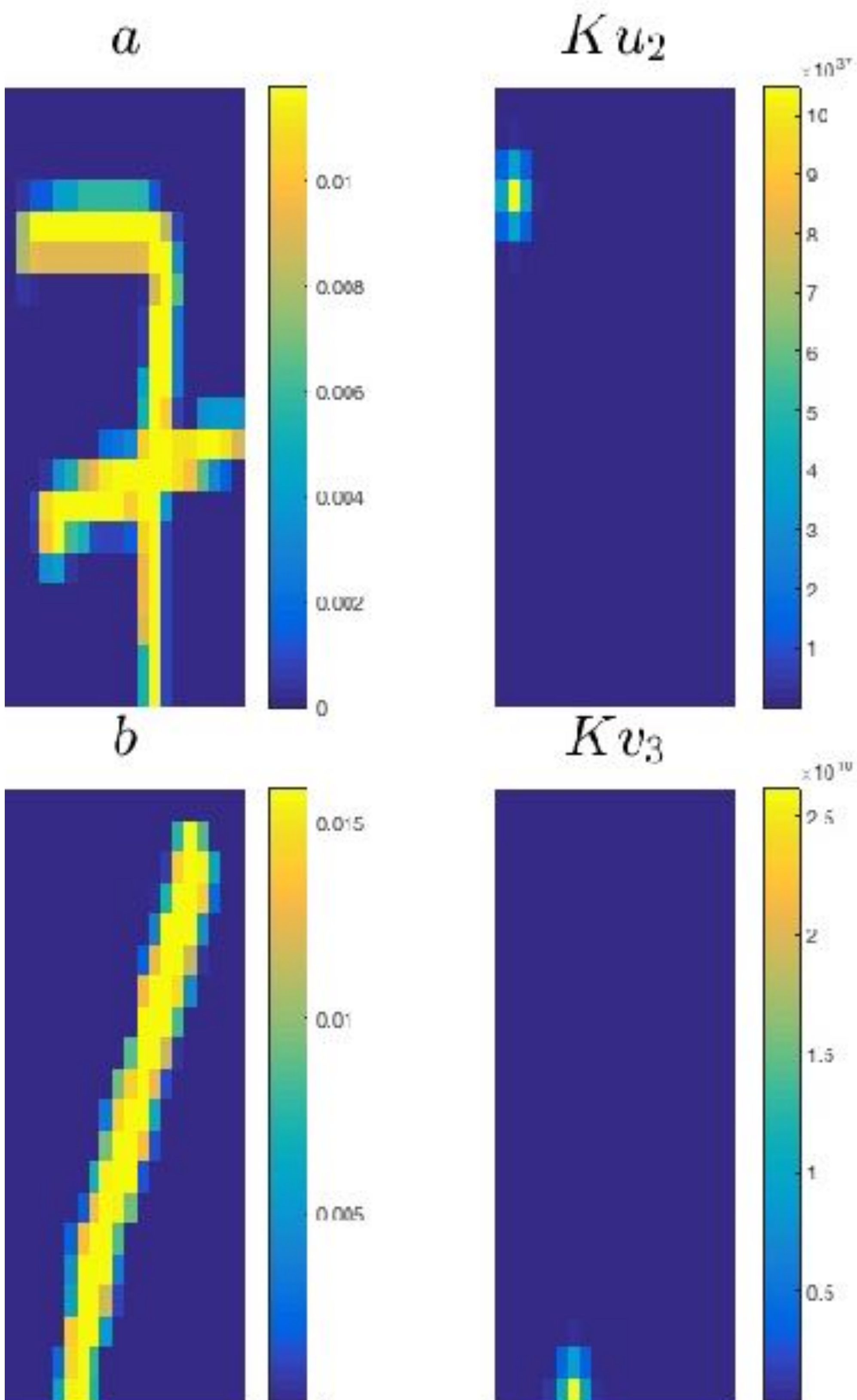
Very Fast EMD Approx. Solver



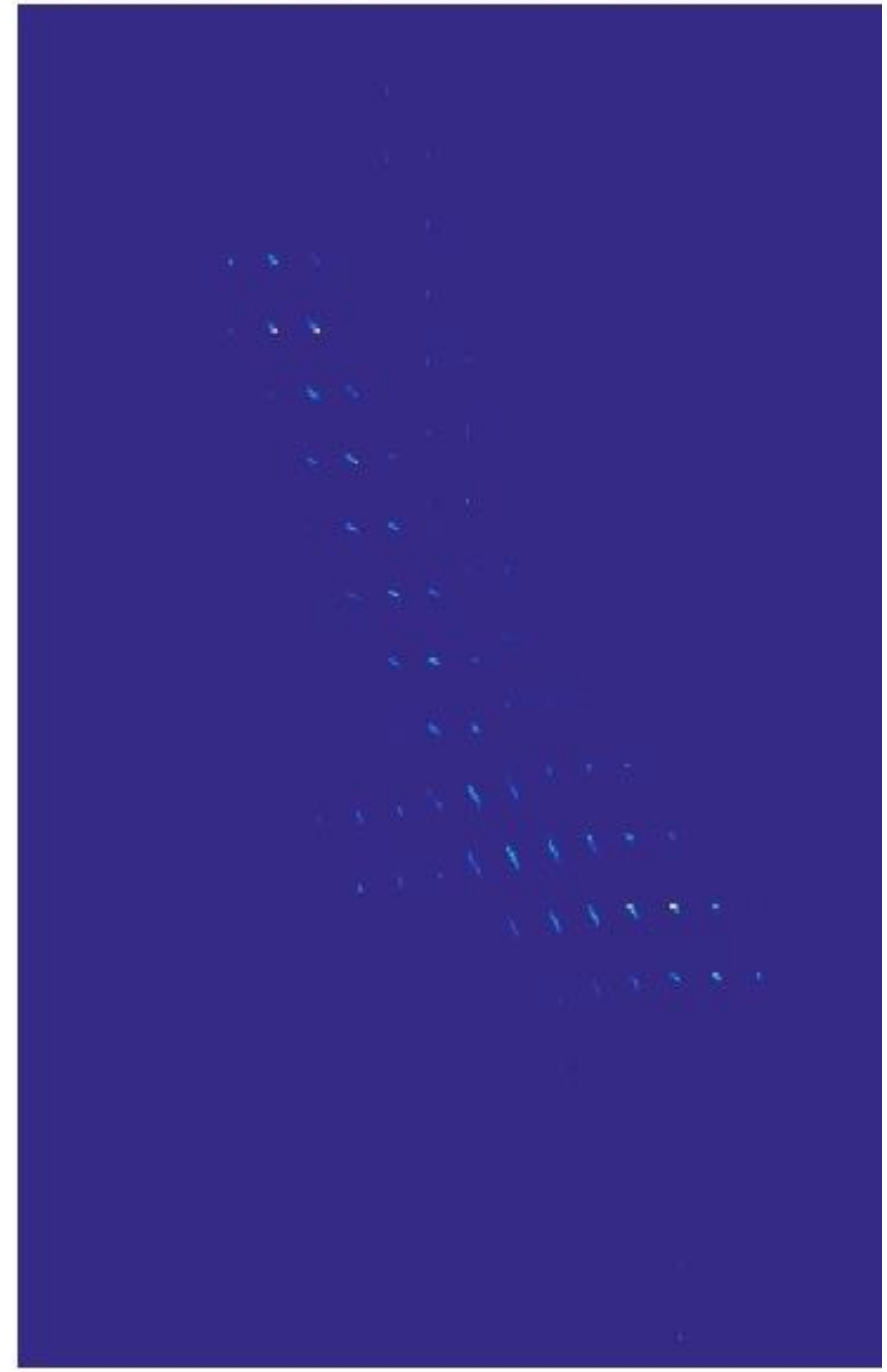
Very Fast EMD Approx. Solver



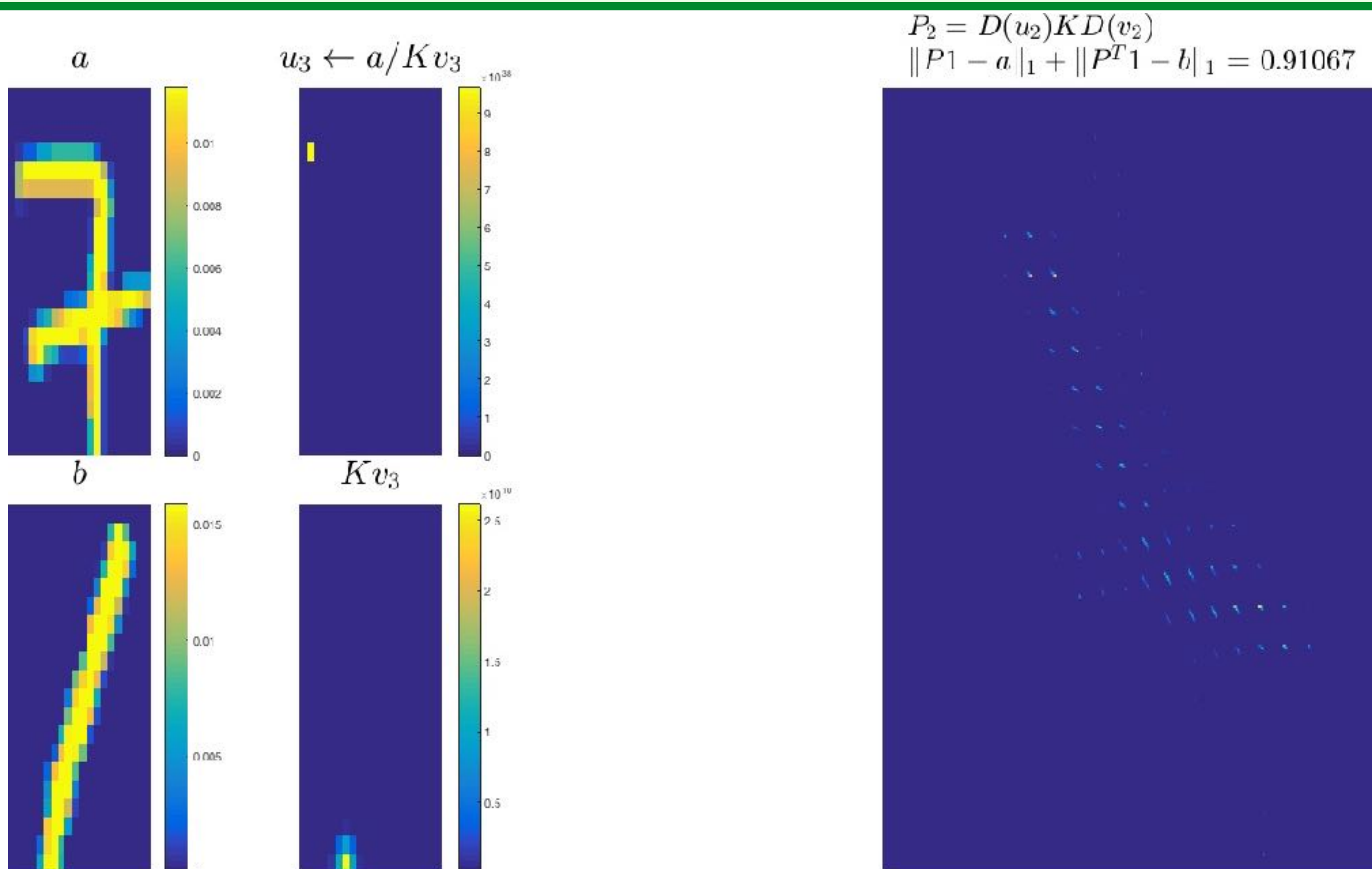
Very Fast EMD Approx. Solver



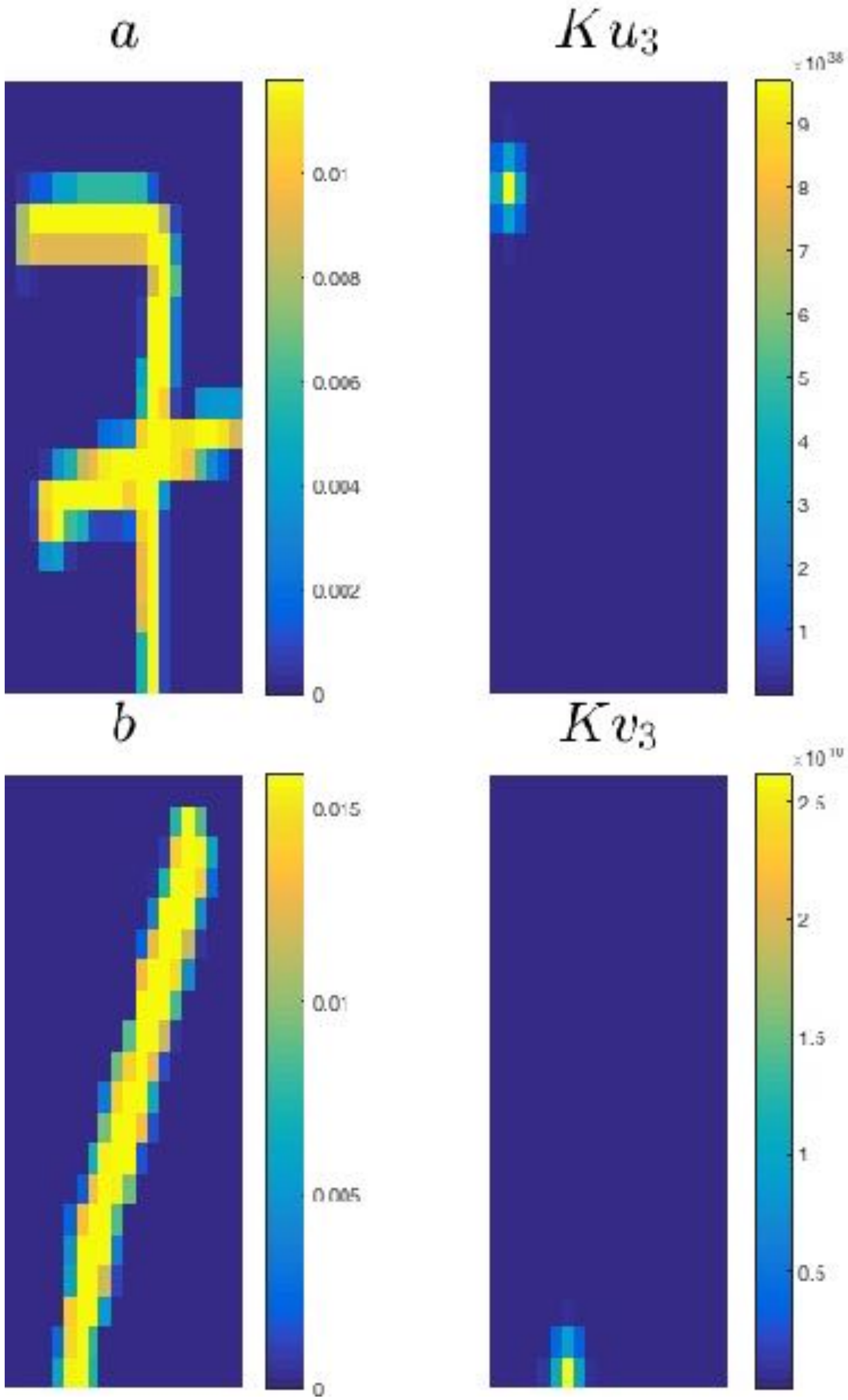
$P_2 = D(u_2)KD(v_2)$
$\|P\mathbb{1} - a\|_1 + \|P^T\mathbb{1} - b\|_1 = 0.91067$



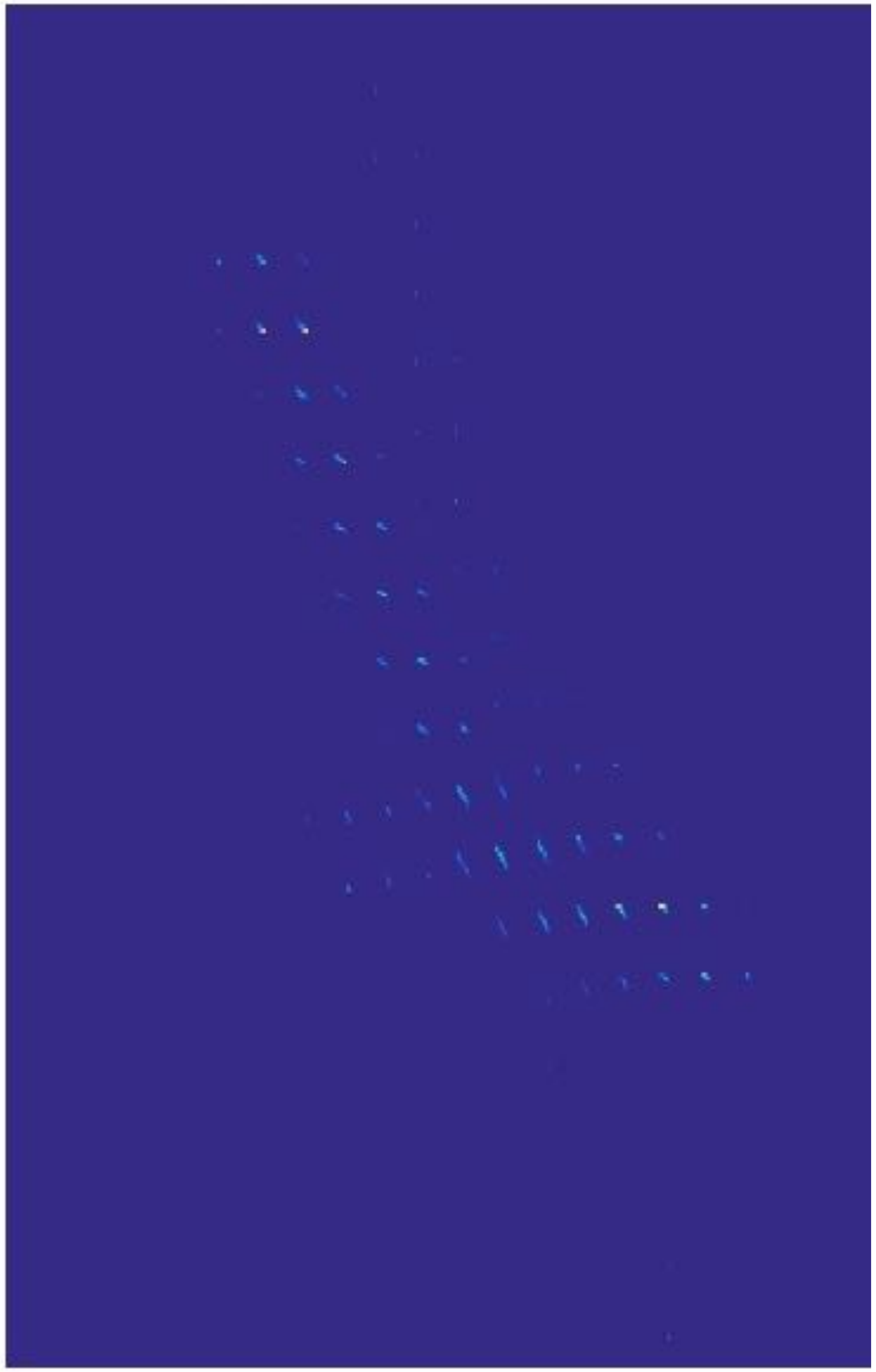
Very Fast EMD Approx. Solver



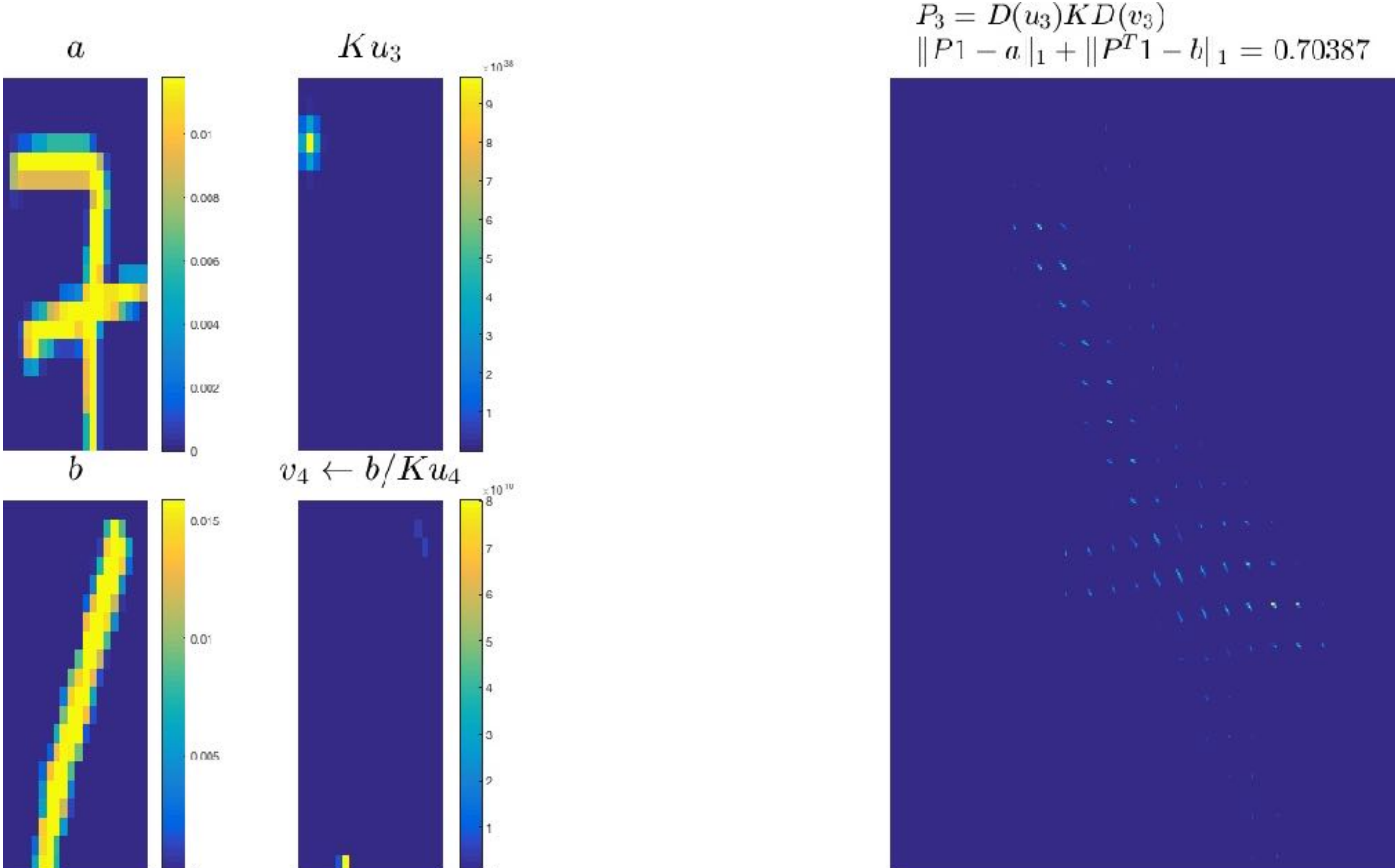
Very Fast EMD Approx. Solver



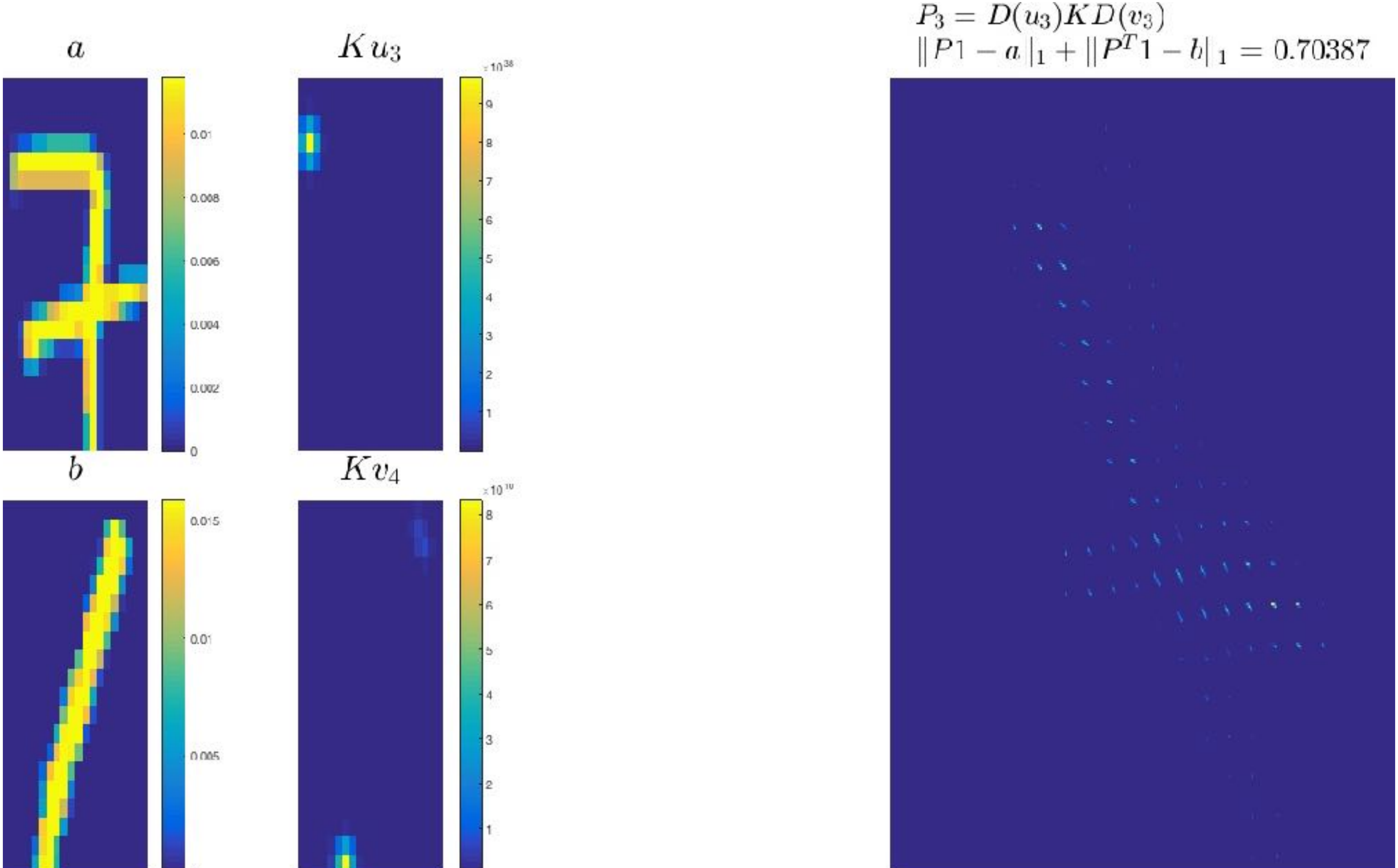
$P_2 = D(u_2)KD(v_2)$
$\|P\mathbb{1} - a\|_1 + \|P^T\mathbb{1} - b\|_1 = 0.91067$



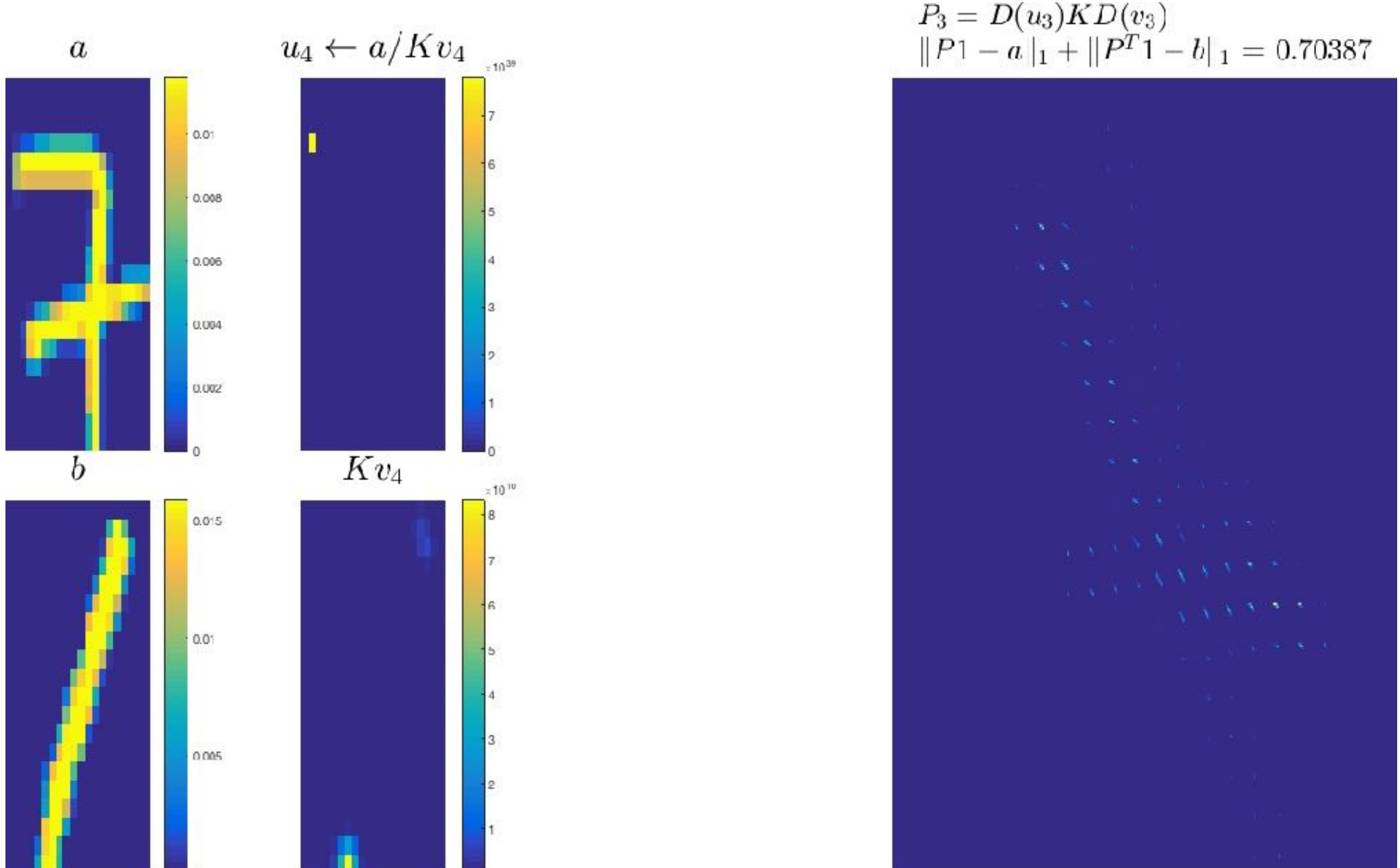
Very Fast EMD Approx. Solver



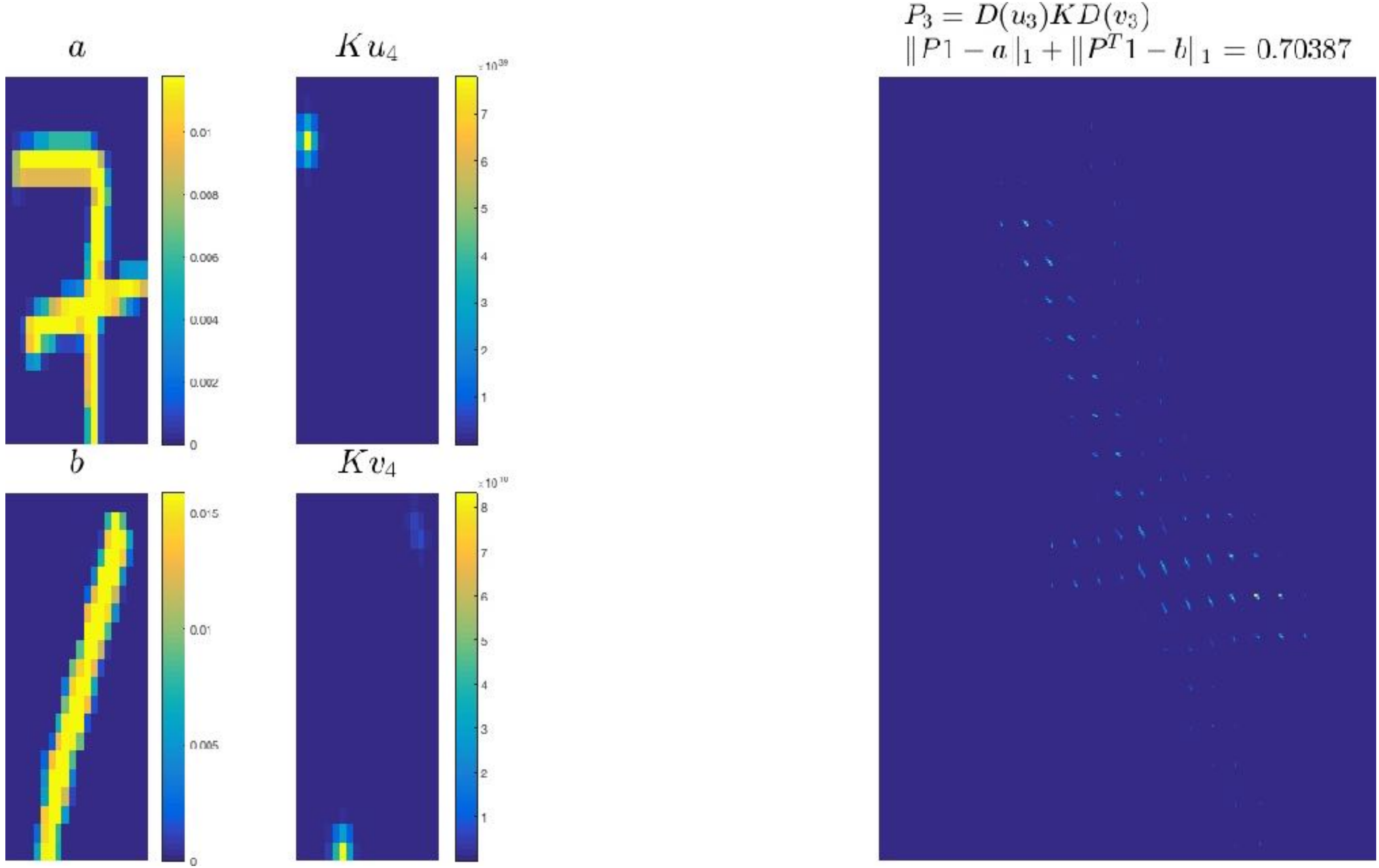
Very Fast EMD Approx. Solver



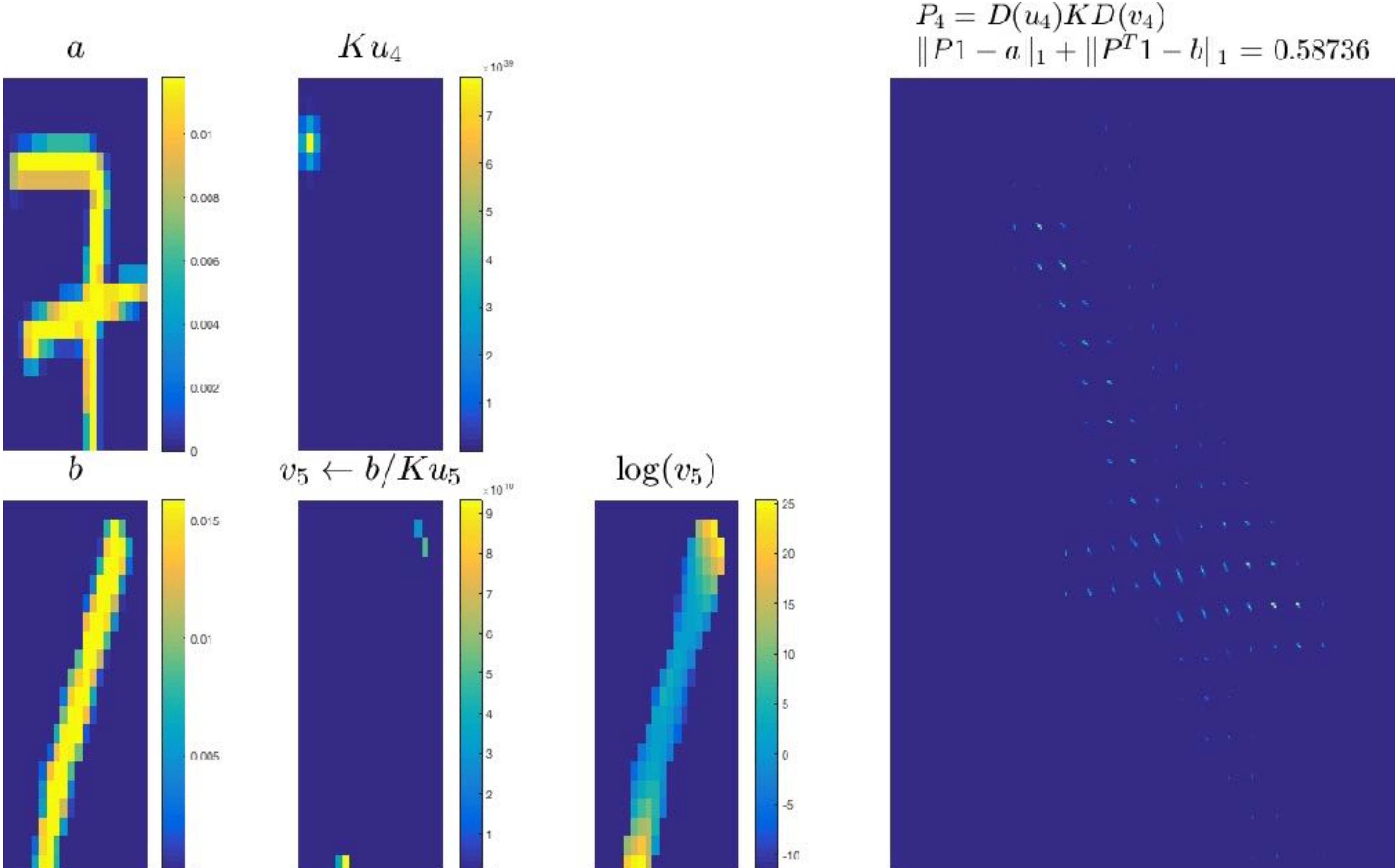
Very Fast EMD Approx. Solver



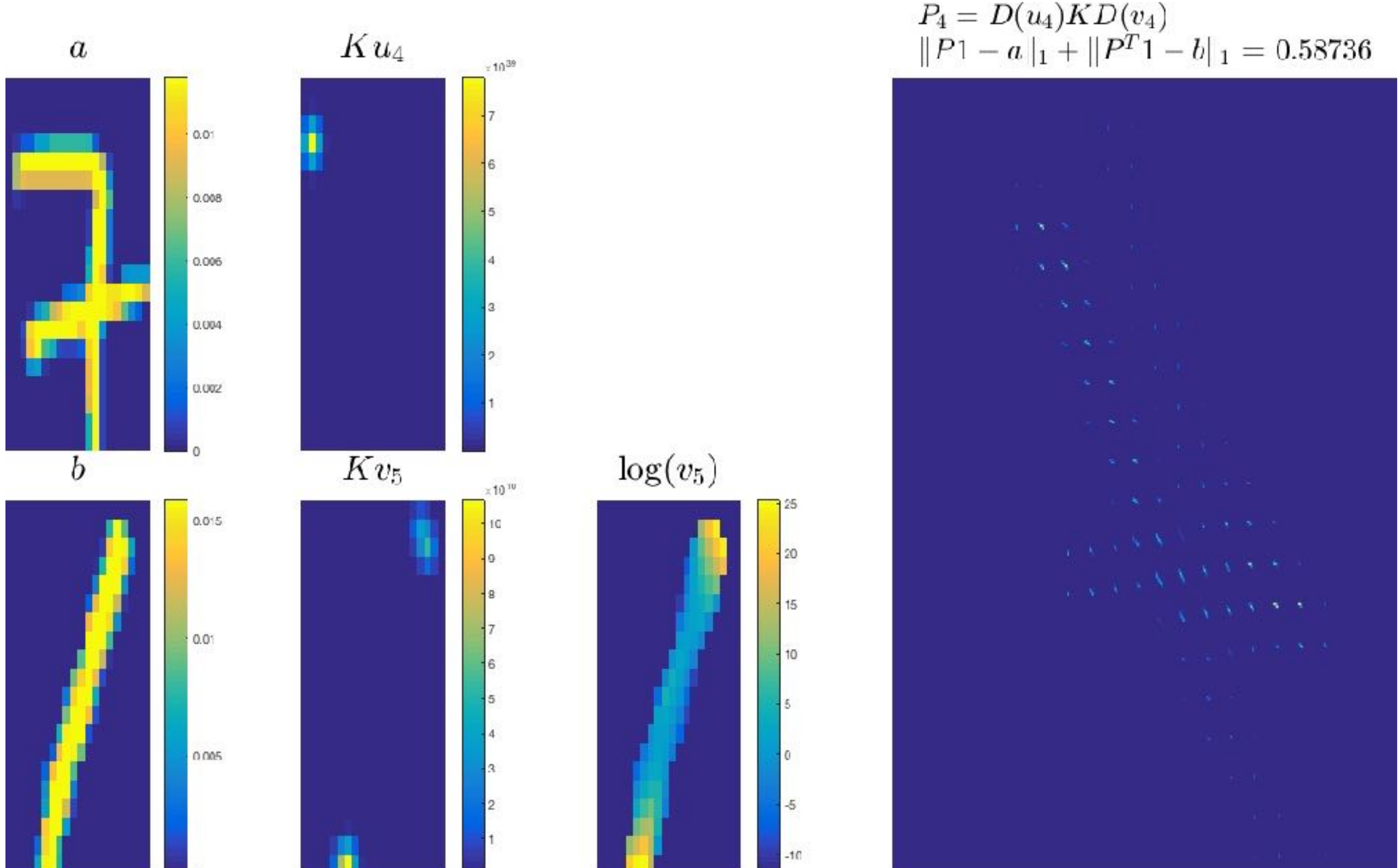
Very Fast EMD Approx. Solver



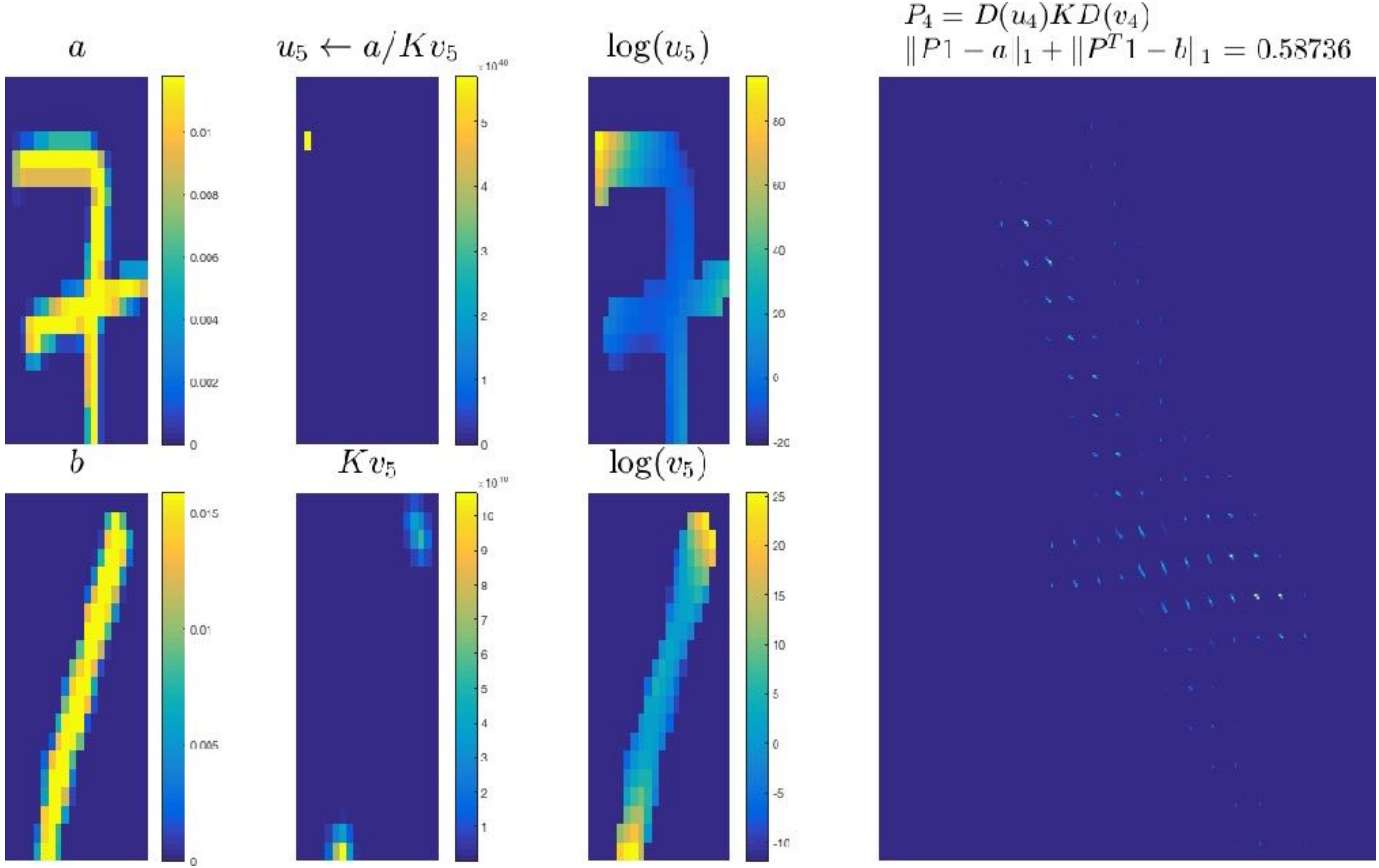
Very Fast EMD Approx. Solver



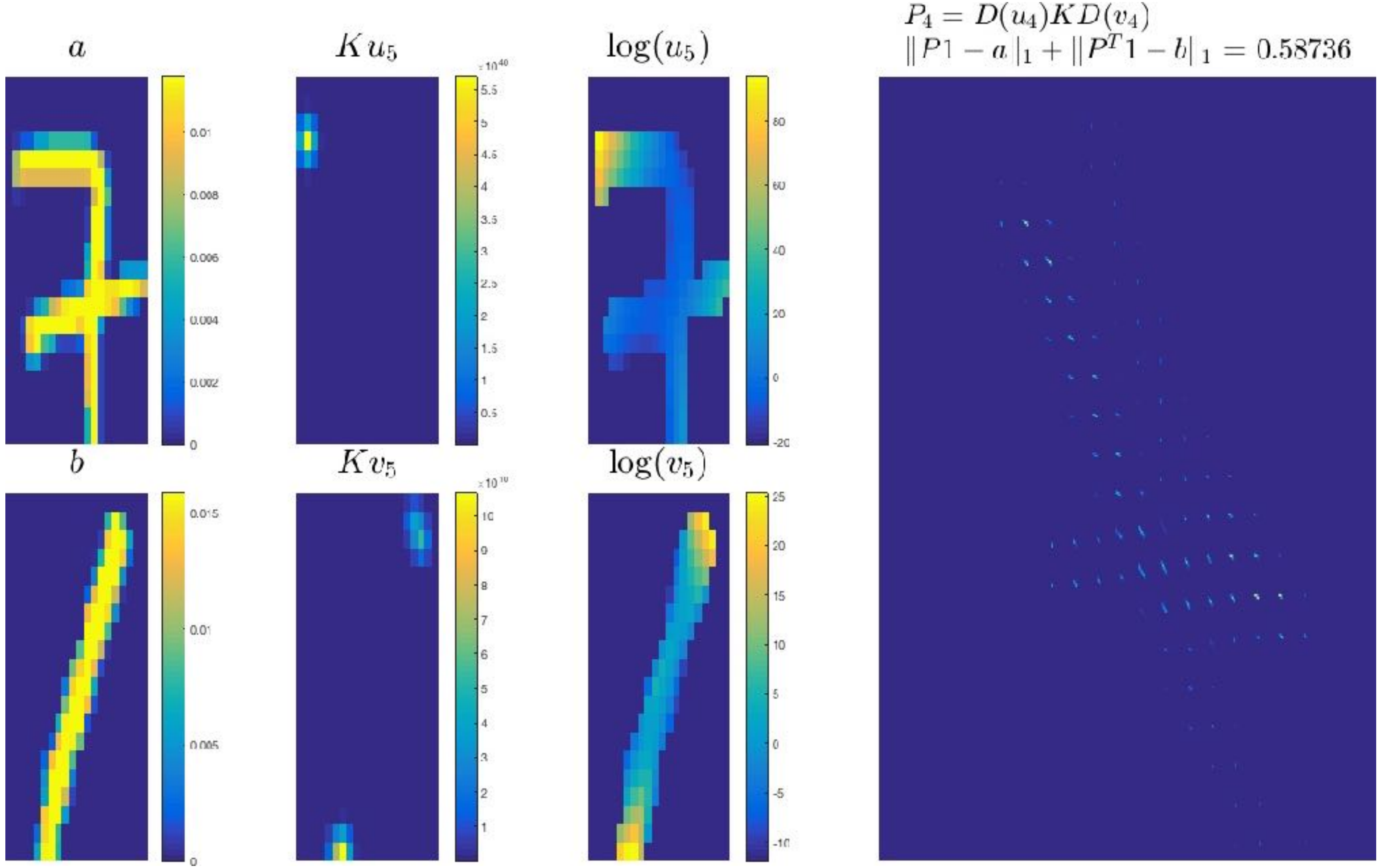
Very Fast EMD Approx. Solver



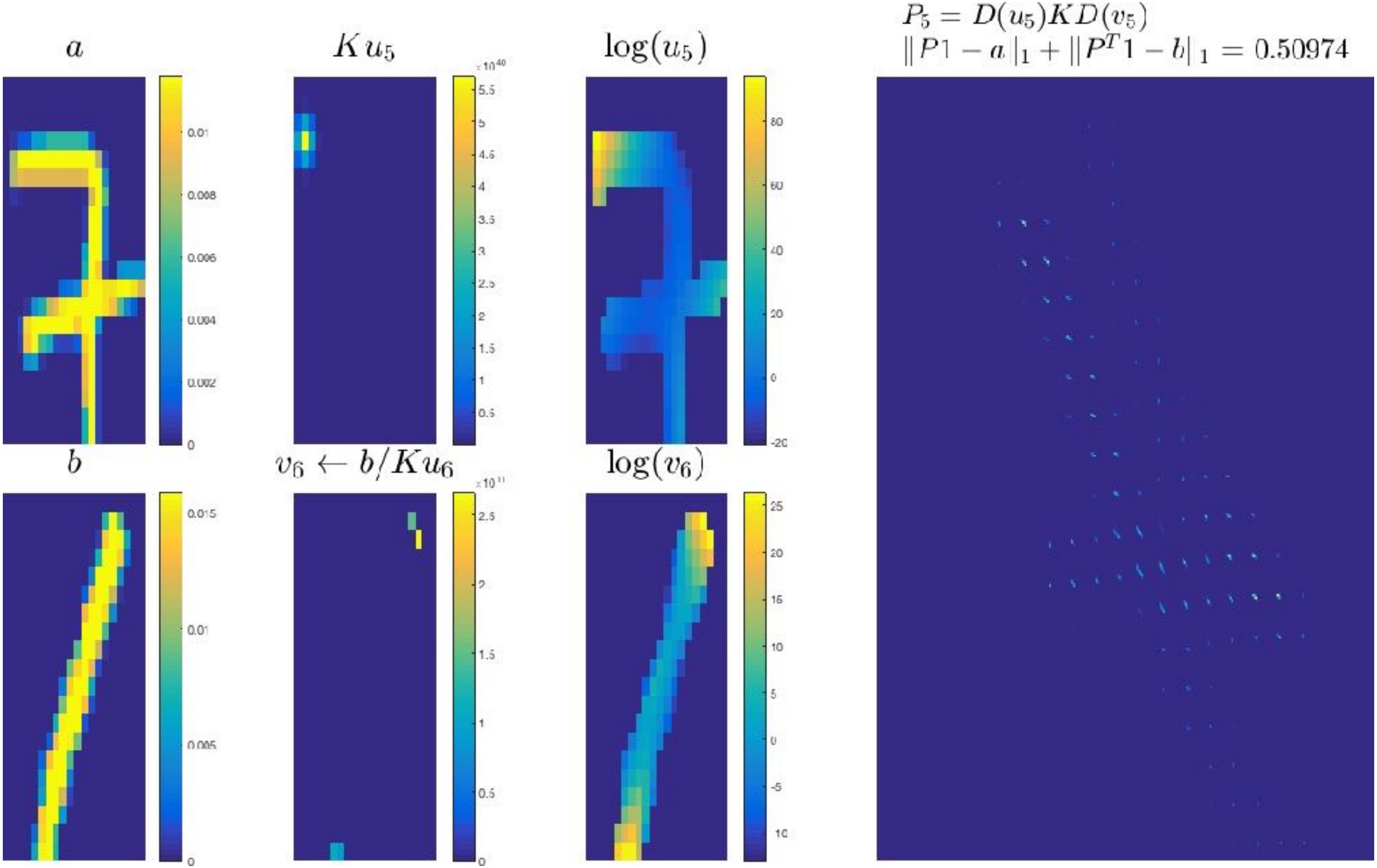
Very Fast EMD Approx. Solver



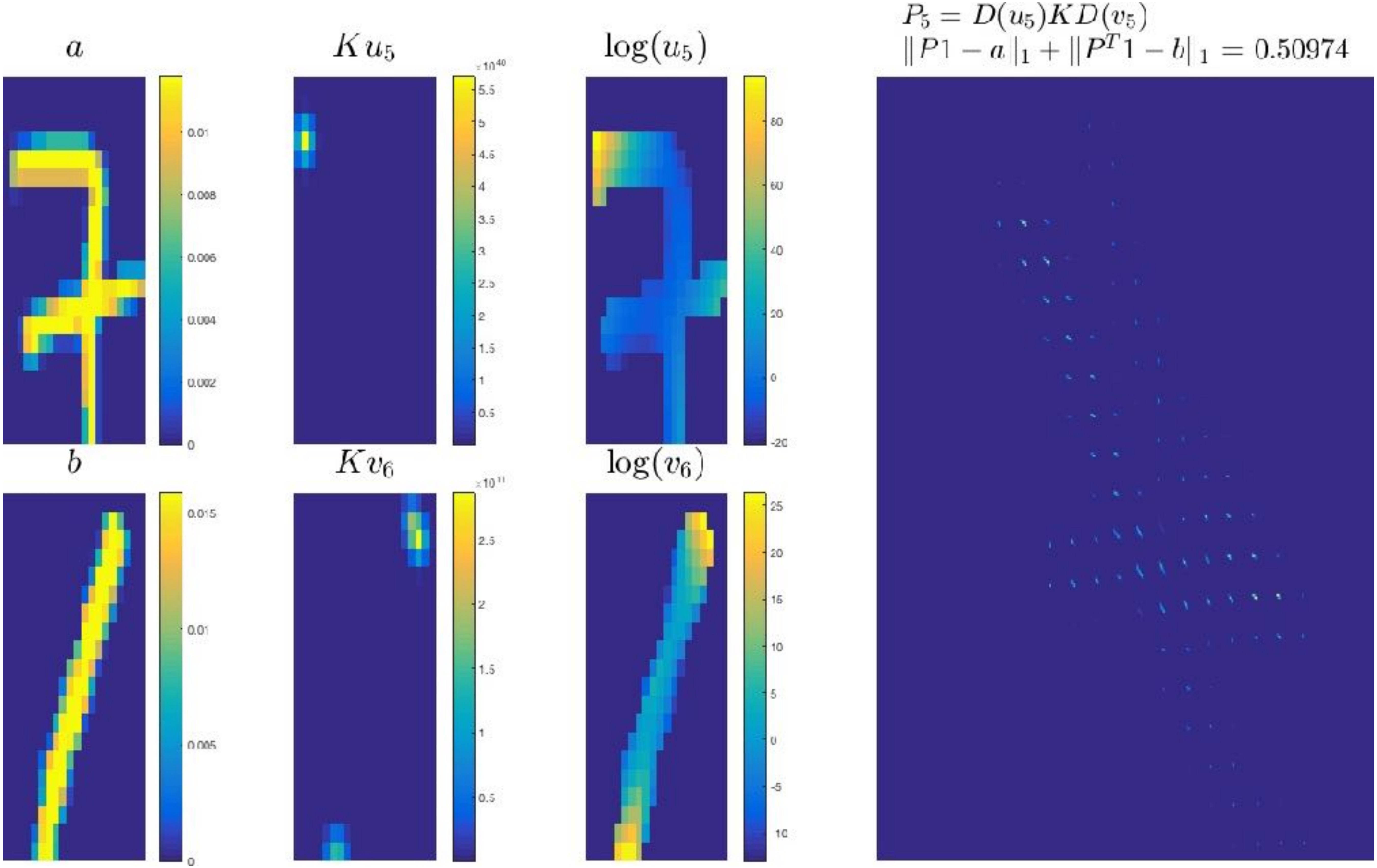
Very Fast EMD Approx. Solver



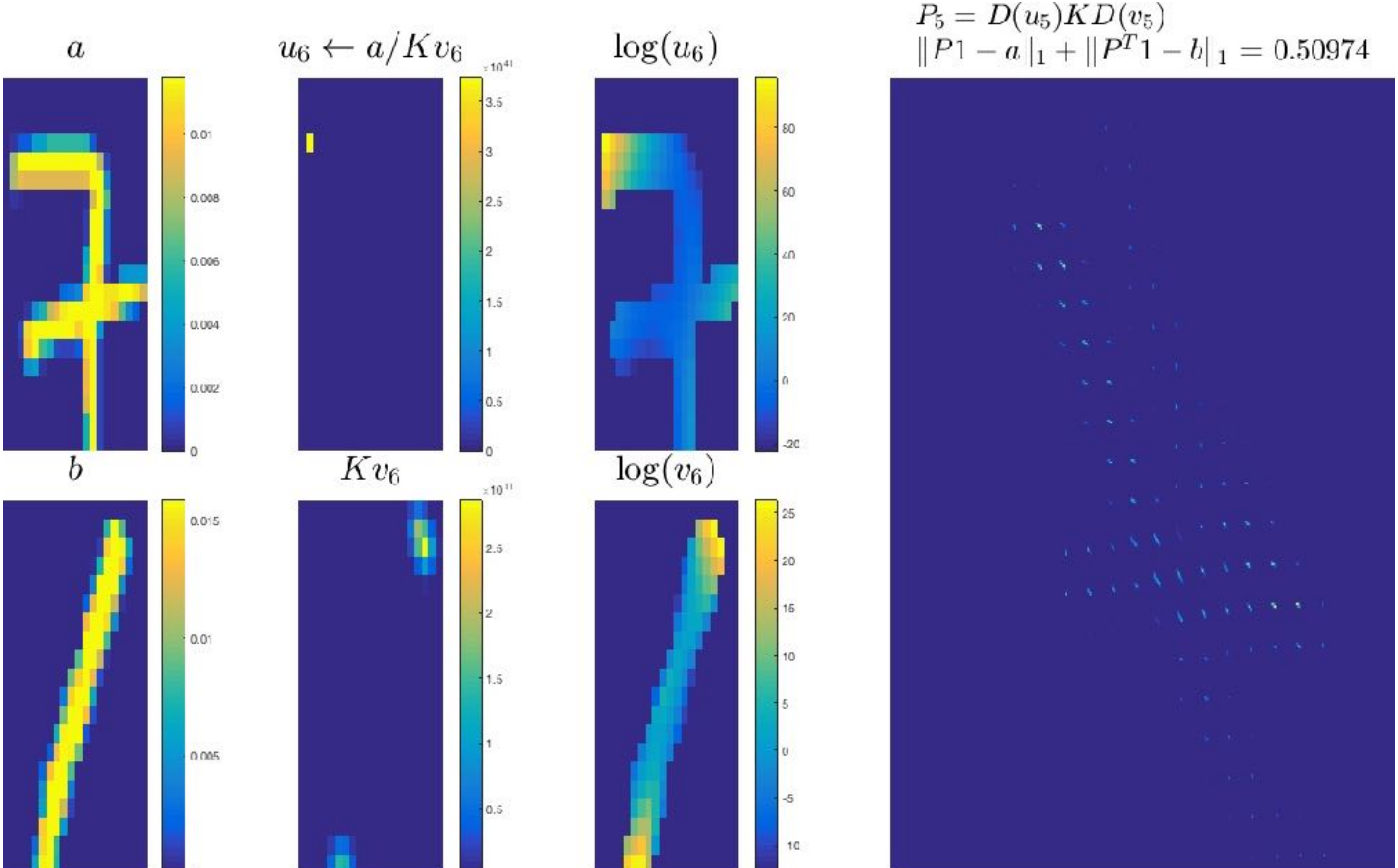
Very Fast EMD Approx. Solver



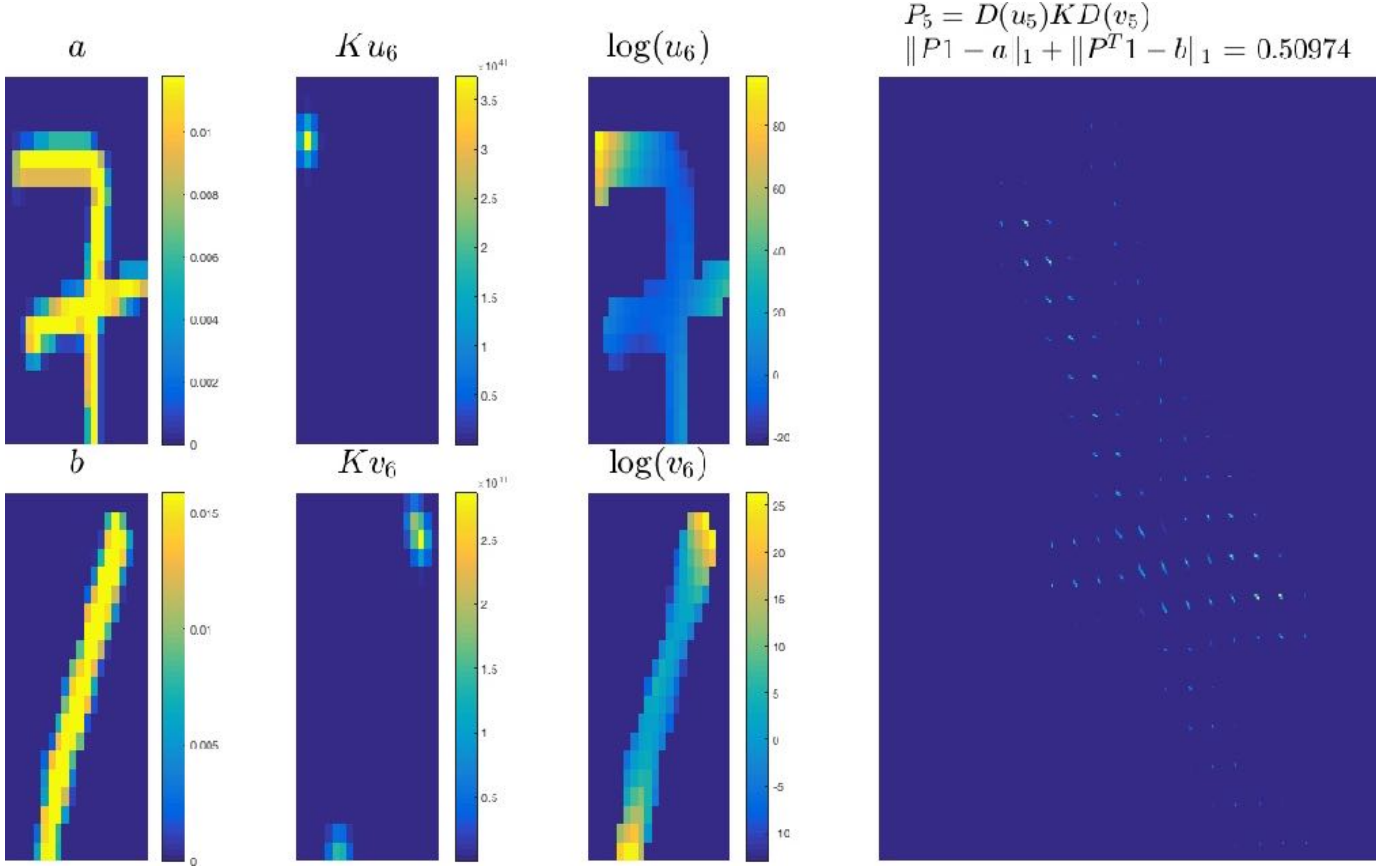
Very Fast EMD Approx. Solver



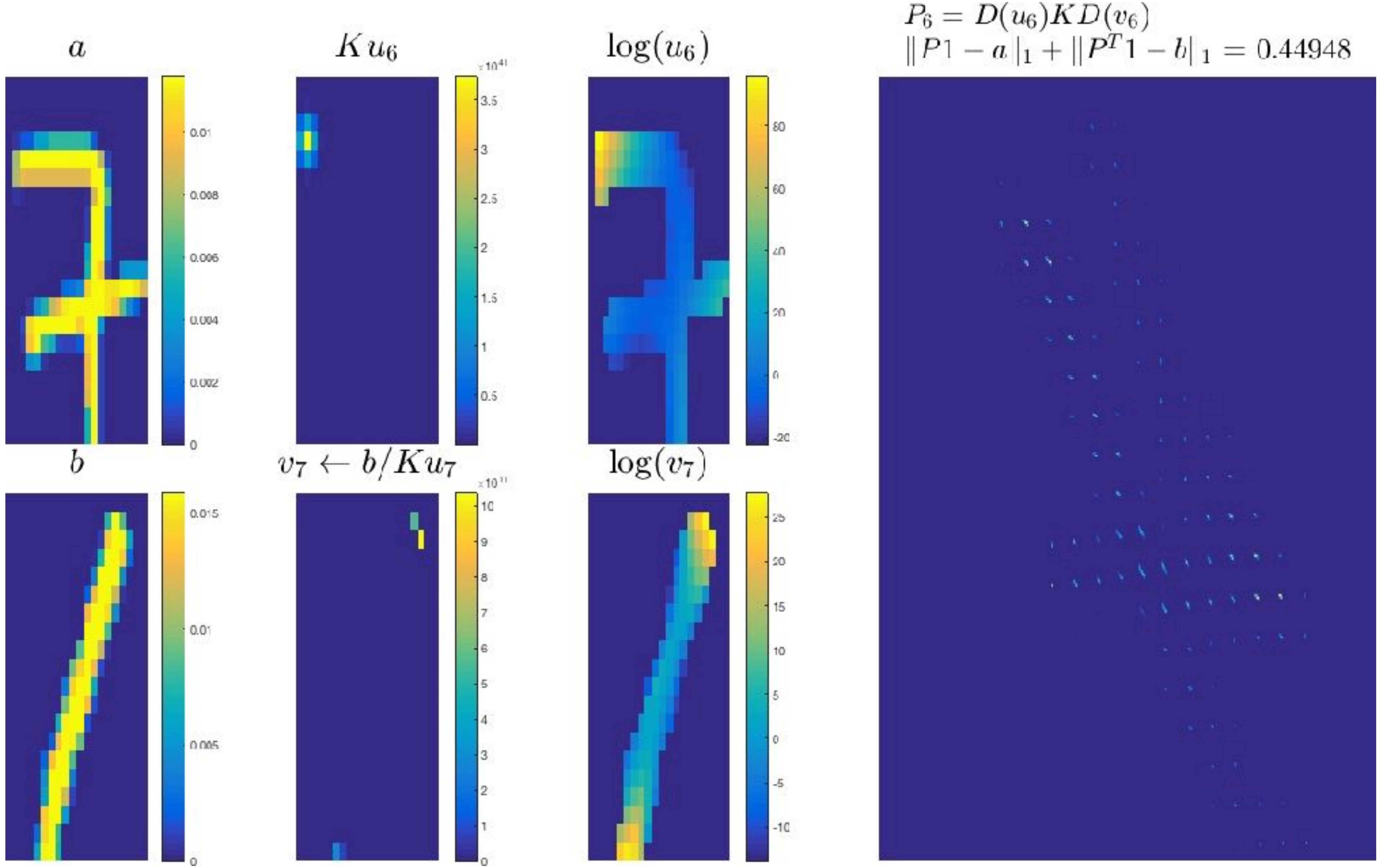
Very Fast EMD Approx. Solver



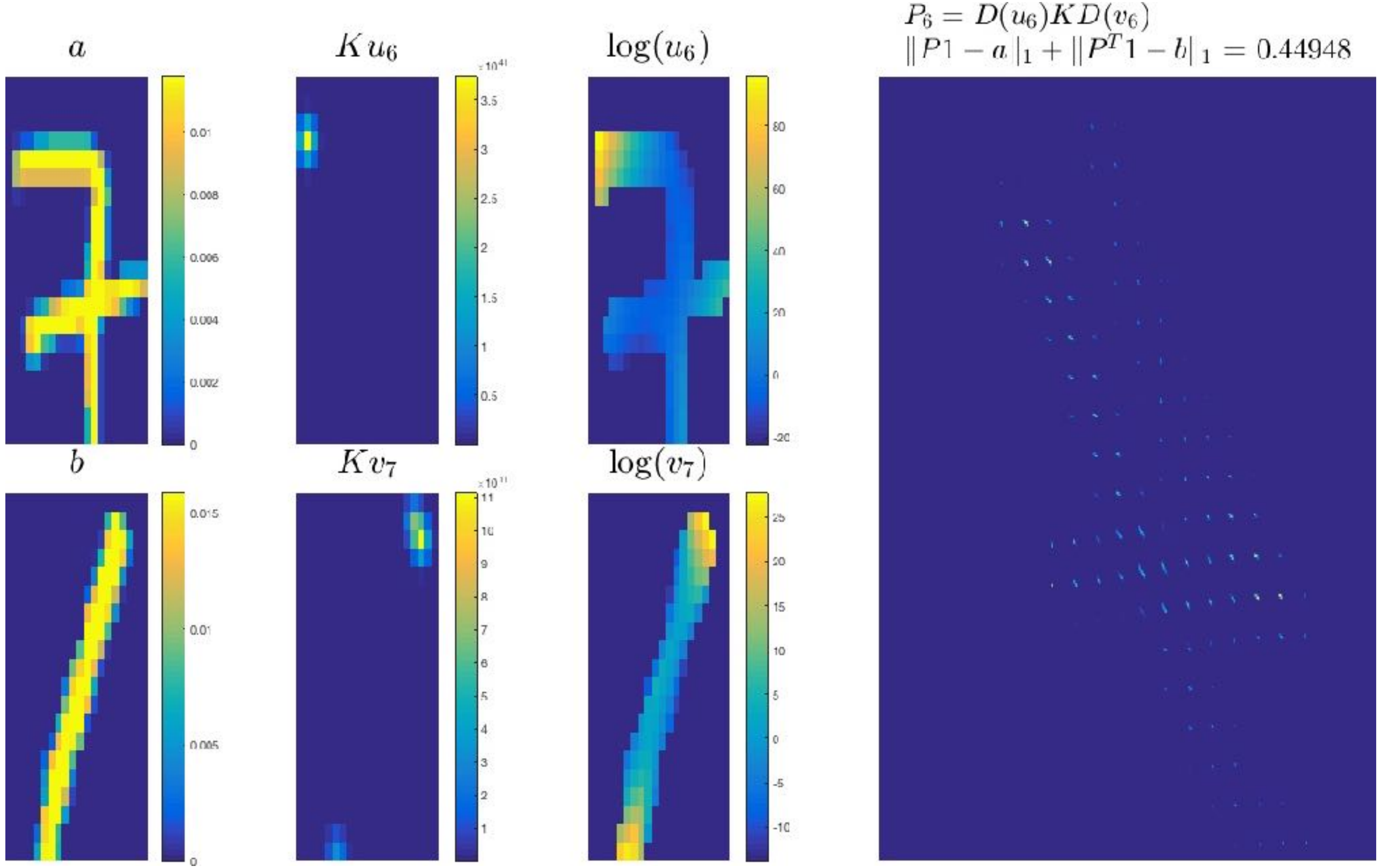
Very Fast EMD Approx. Solver



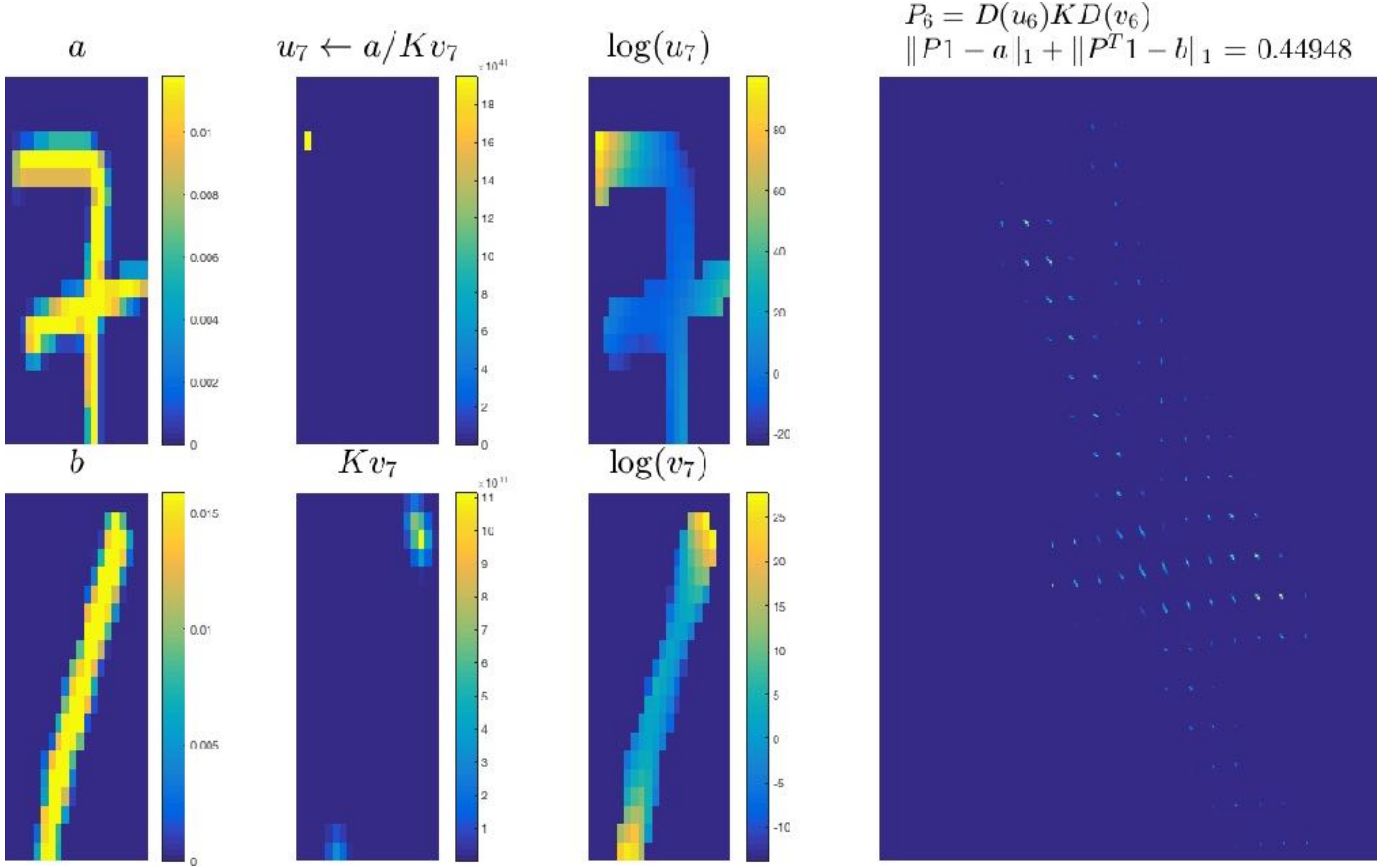
Very Fast EMD Approx. Solver



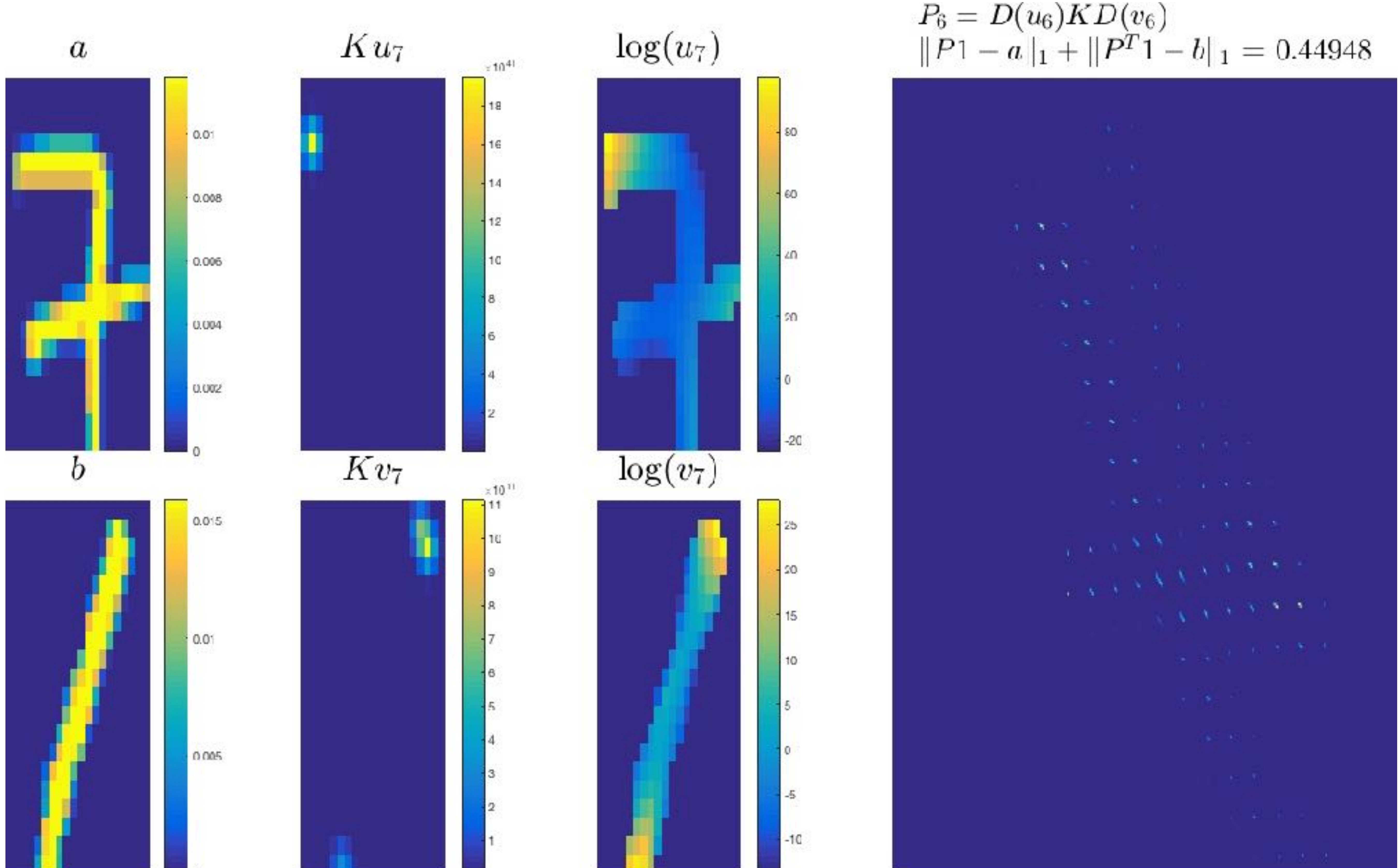
Very Fast EMD Approx. Solver



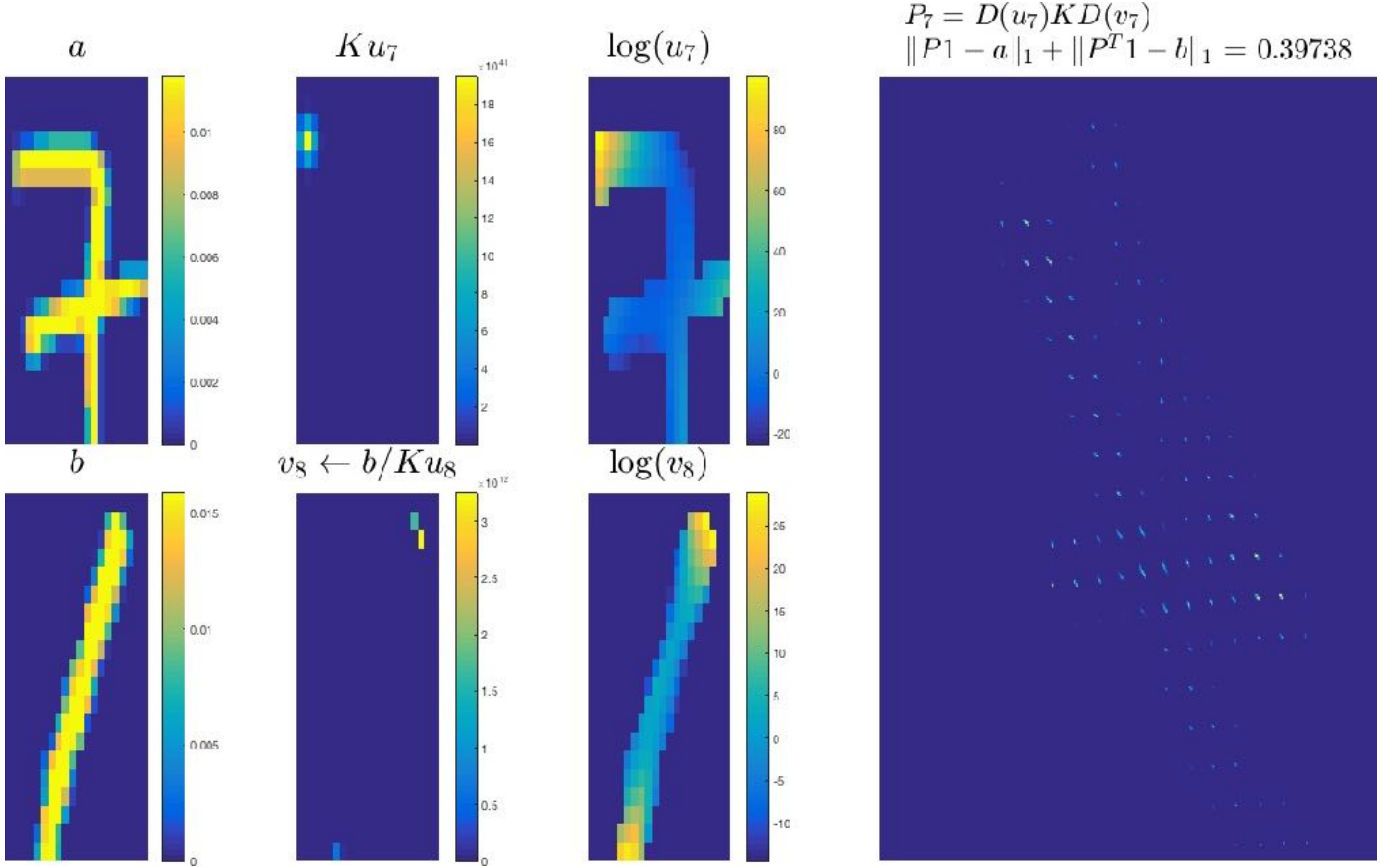
Very Fast EMD Approx. Solver



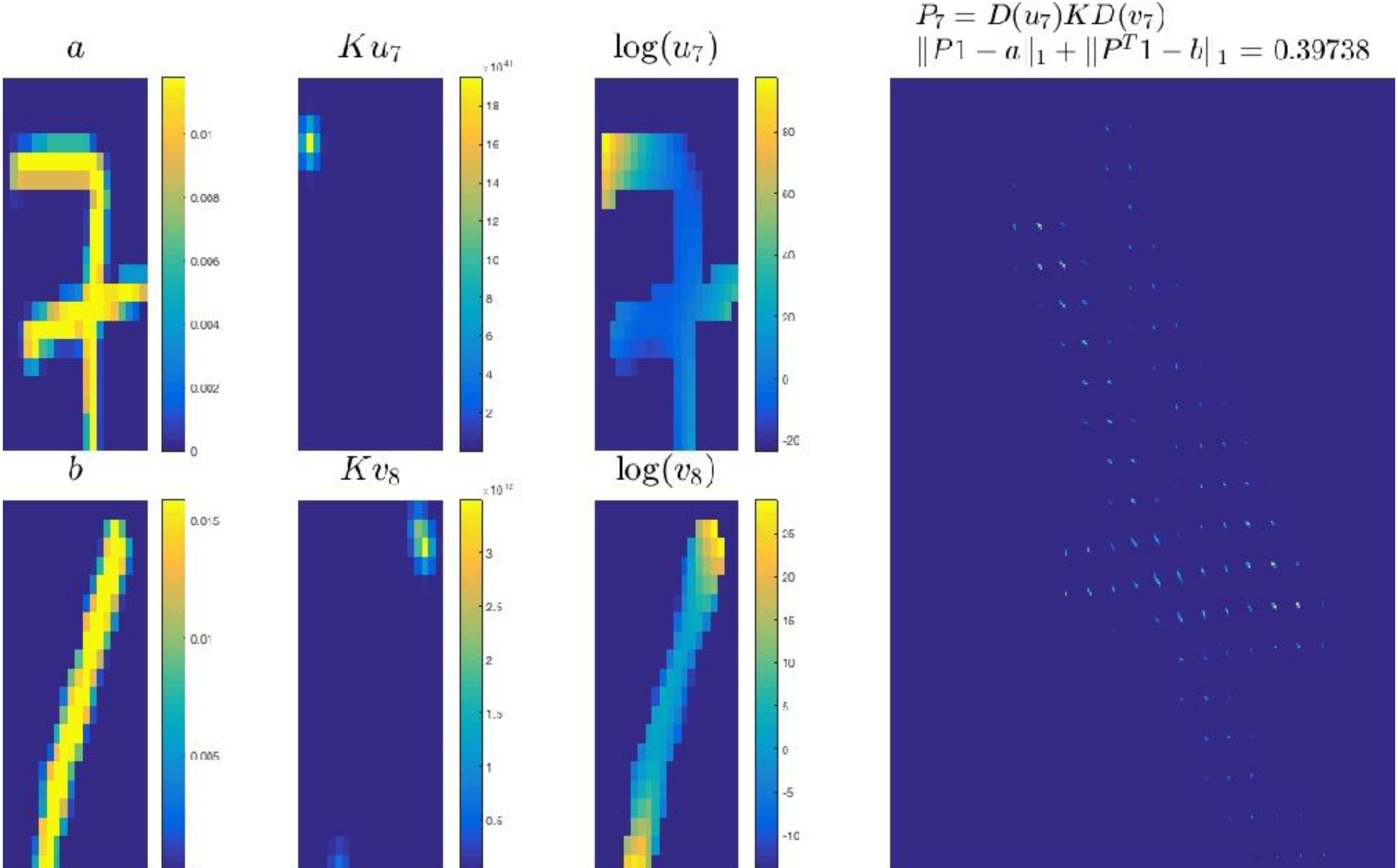
Very Fast EMD Approx. Solver



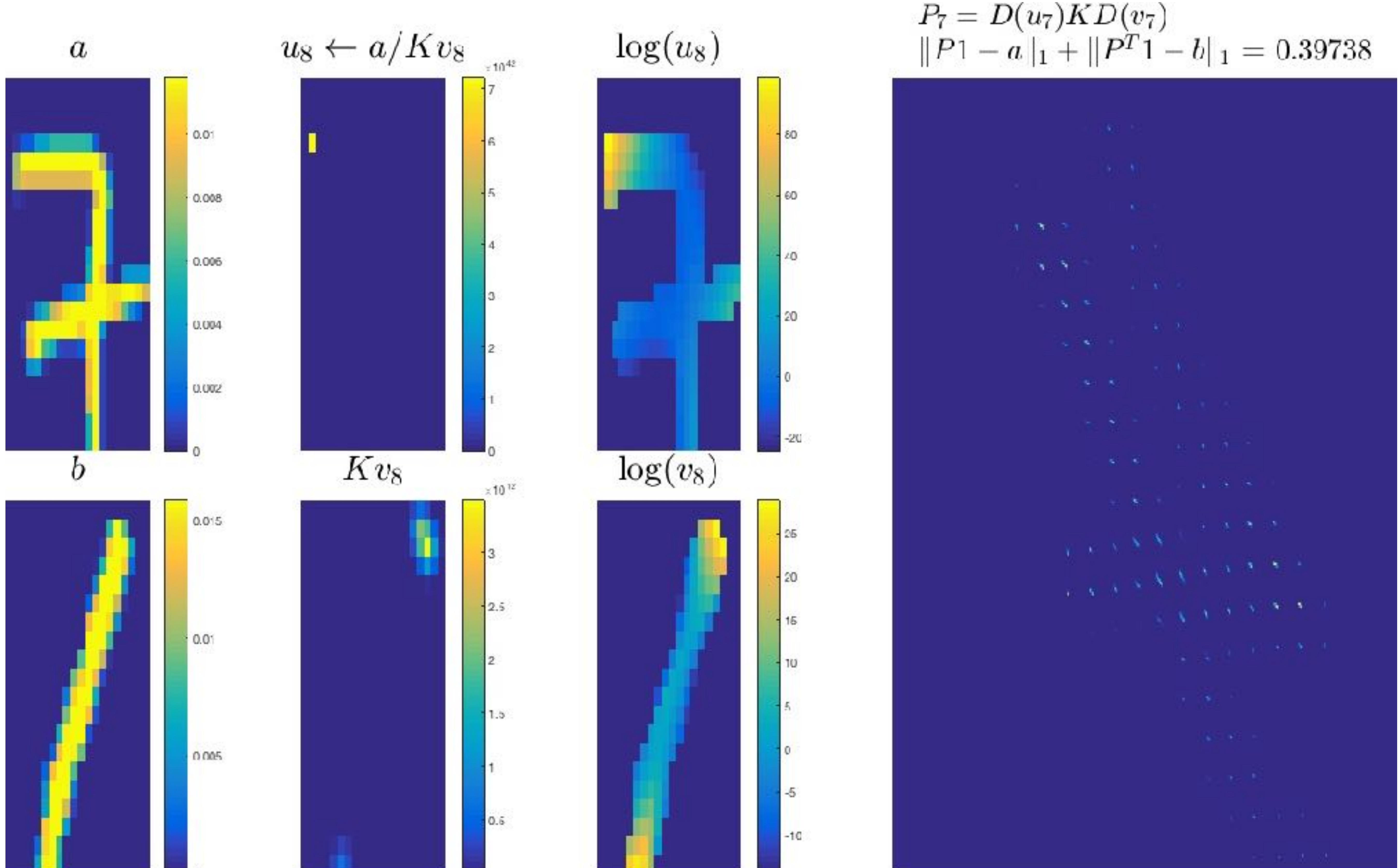
Very Fast EMD Approx. Solver



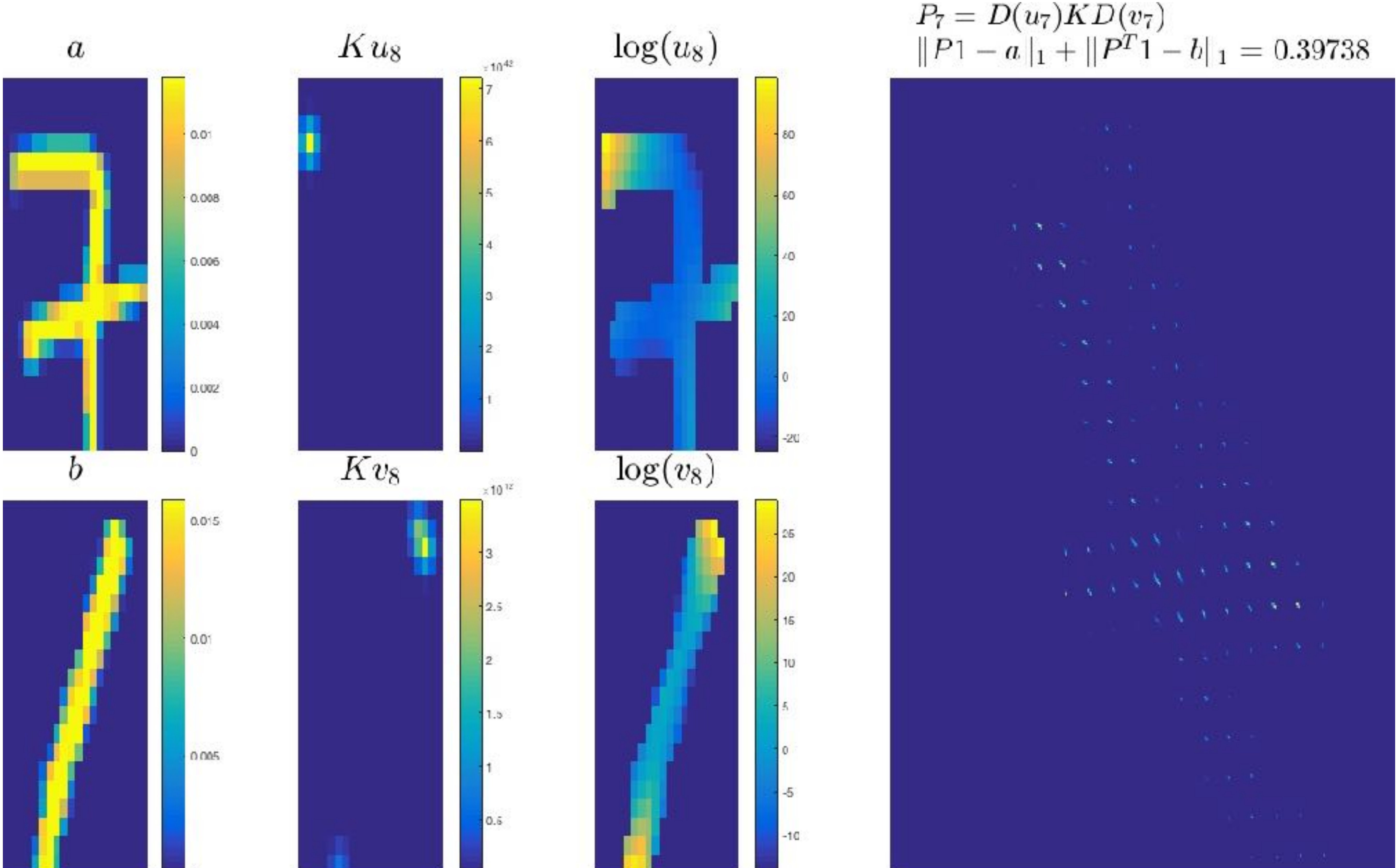
Very Fast EMD Approx. Solver



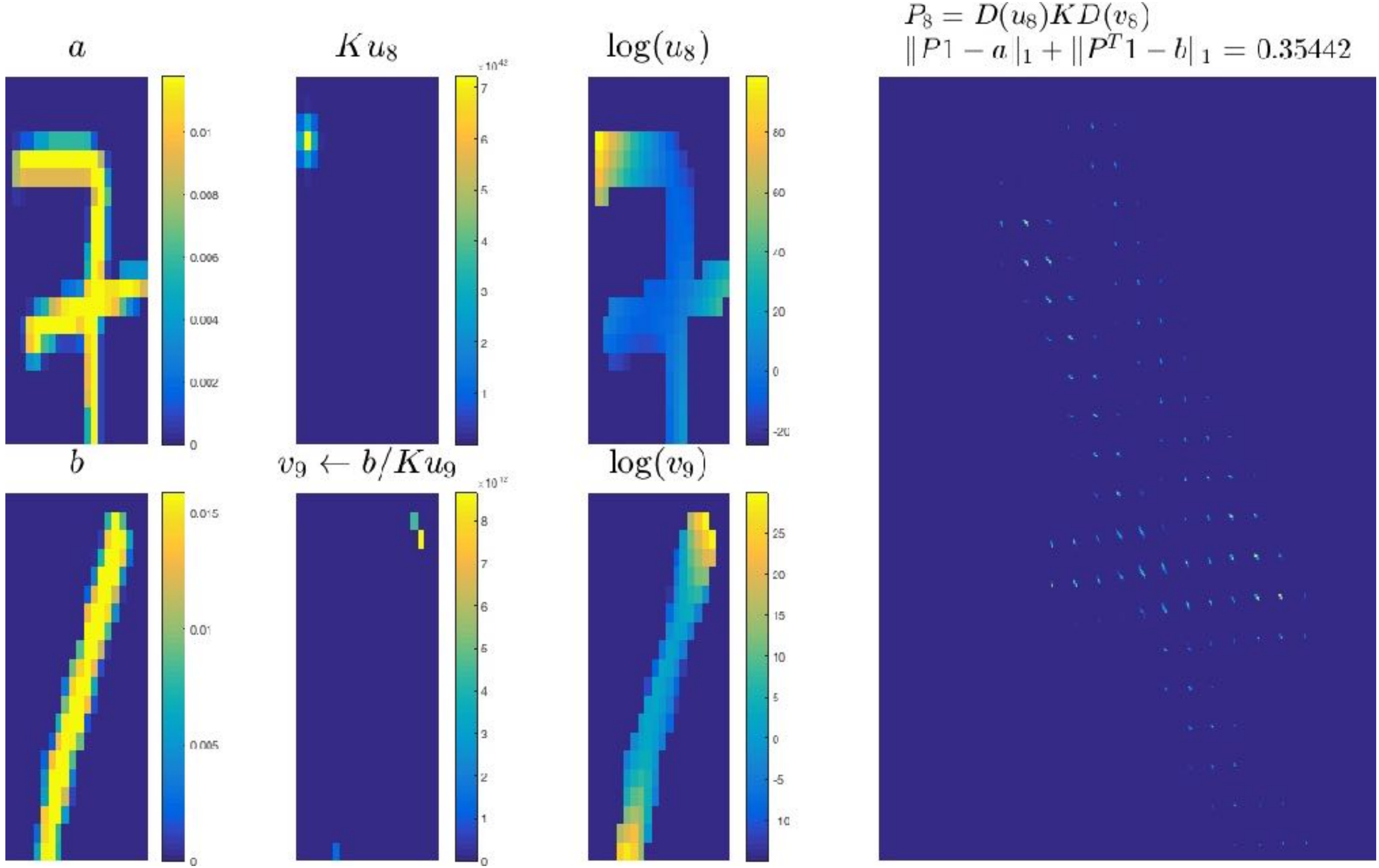
Very Fast EMD Approx. Solver



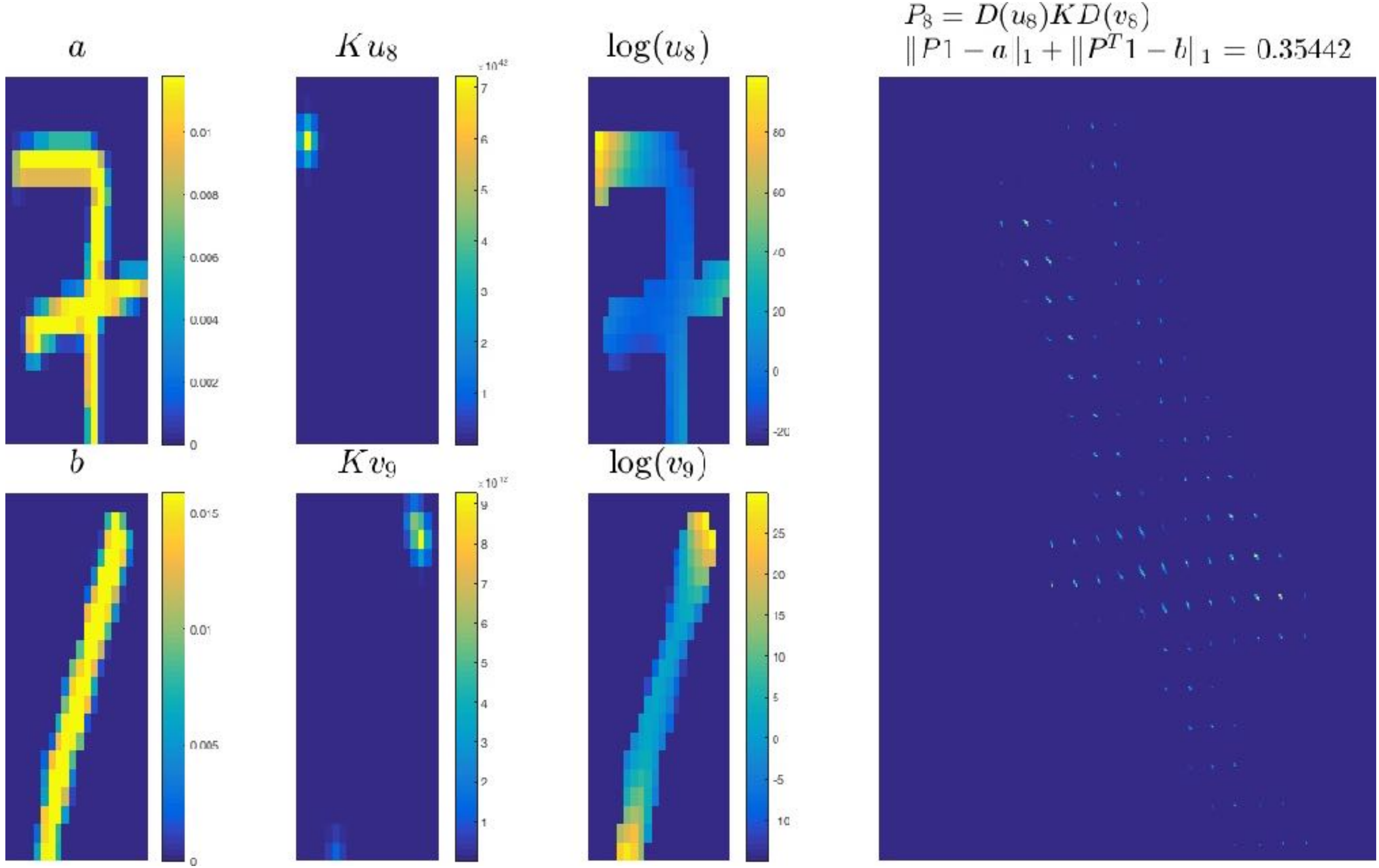
Very Fast EMD Approx. Solver



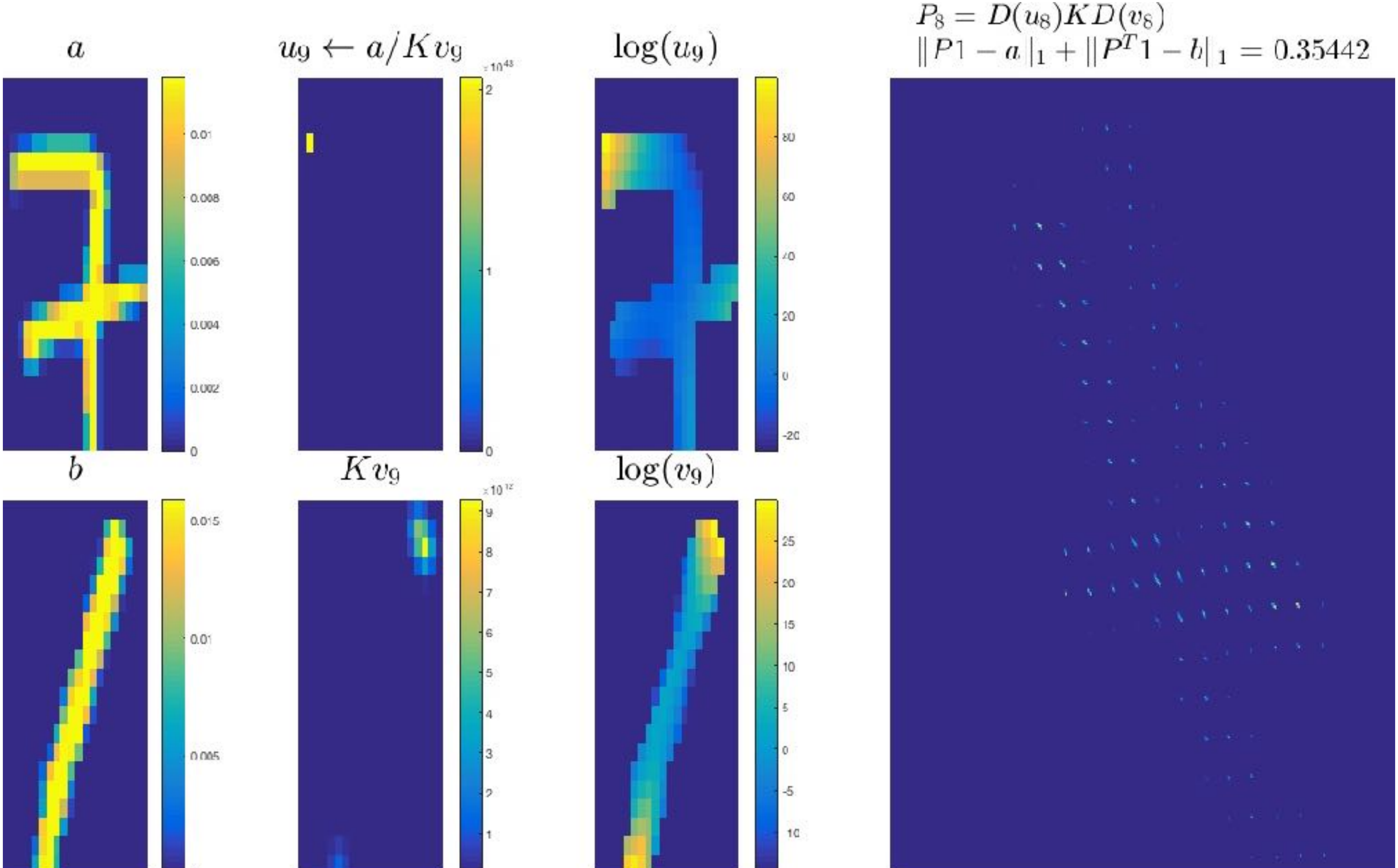
Very Fast EMD Approx. Solver



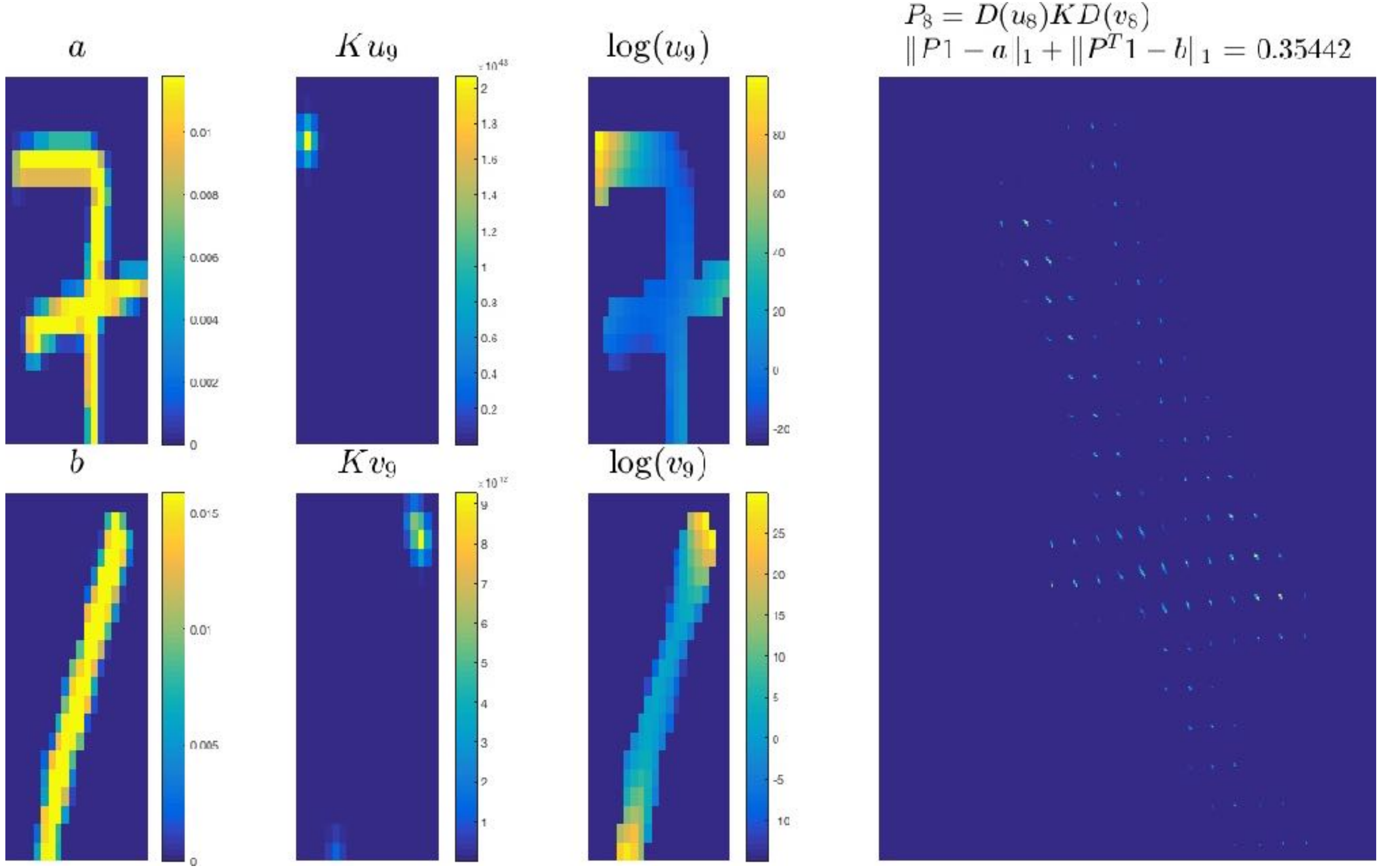
Very Fast EMD Approx. Solver



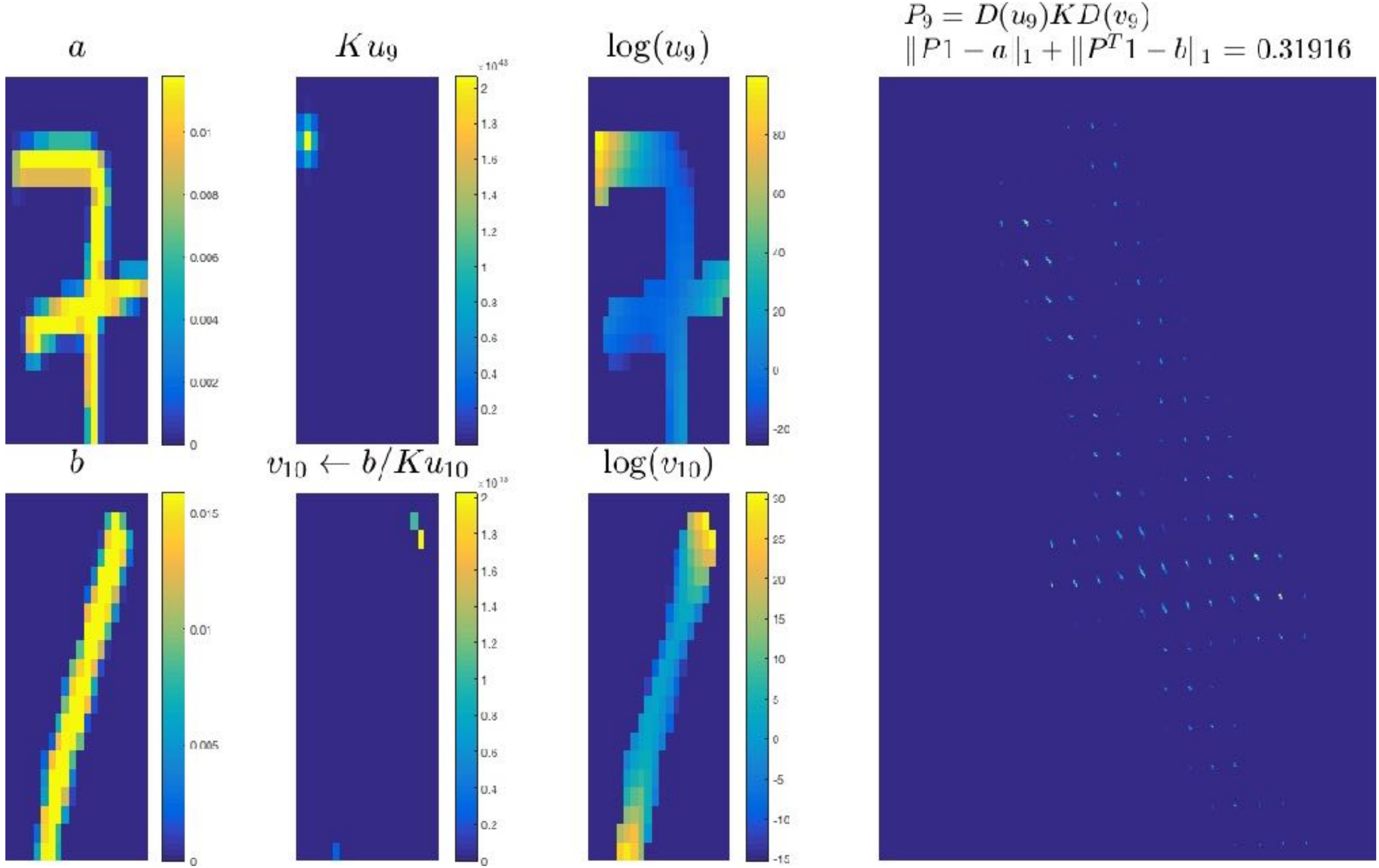
Very Fast EMD Approx. Solver



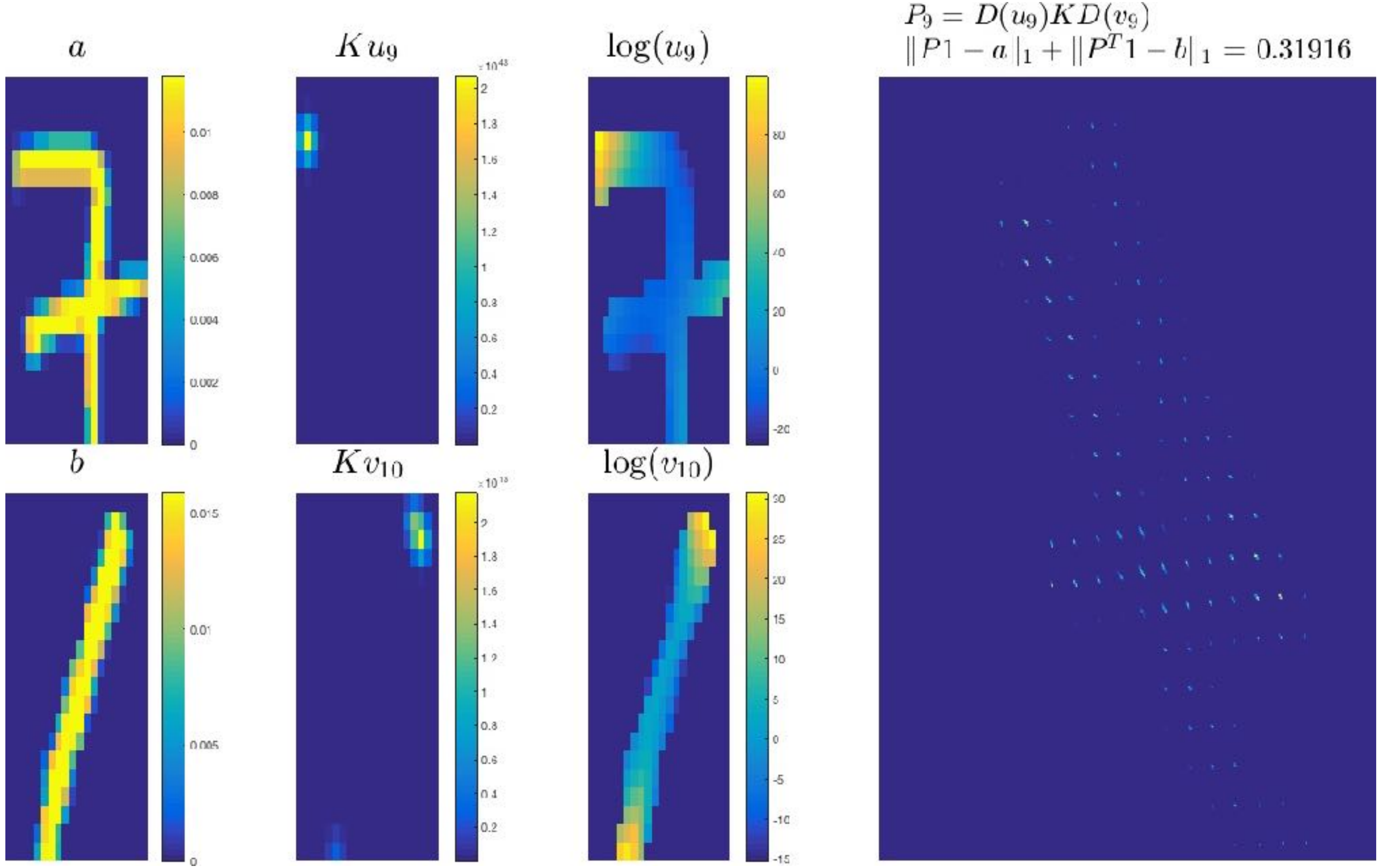
Very Fast EMD Approx. Solver



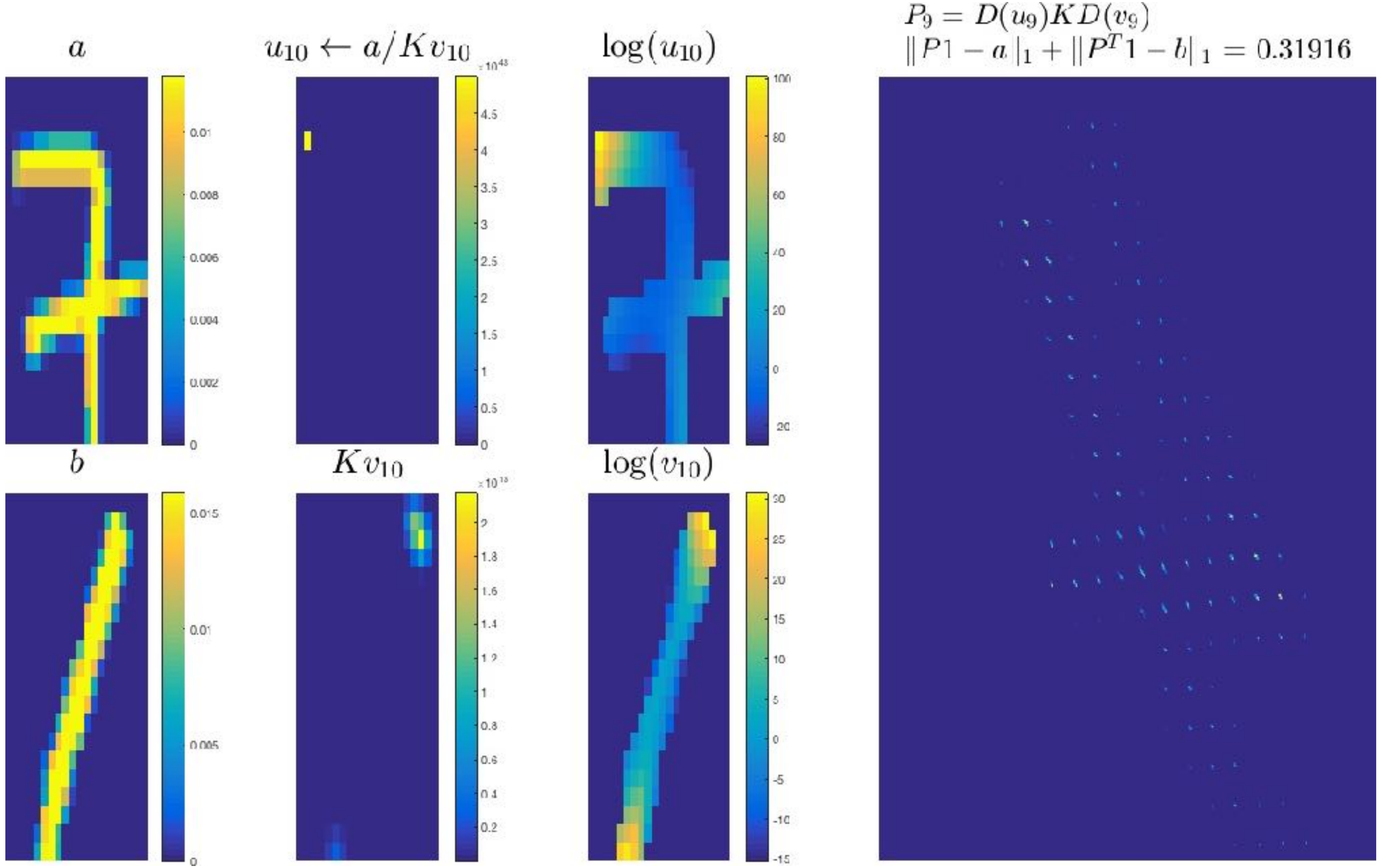
Very Fast EMD Approx. Solver



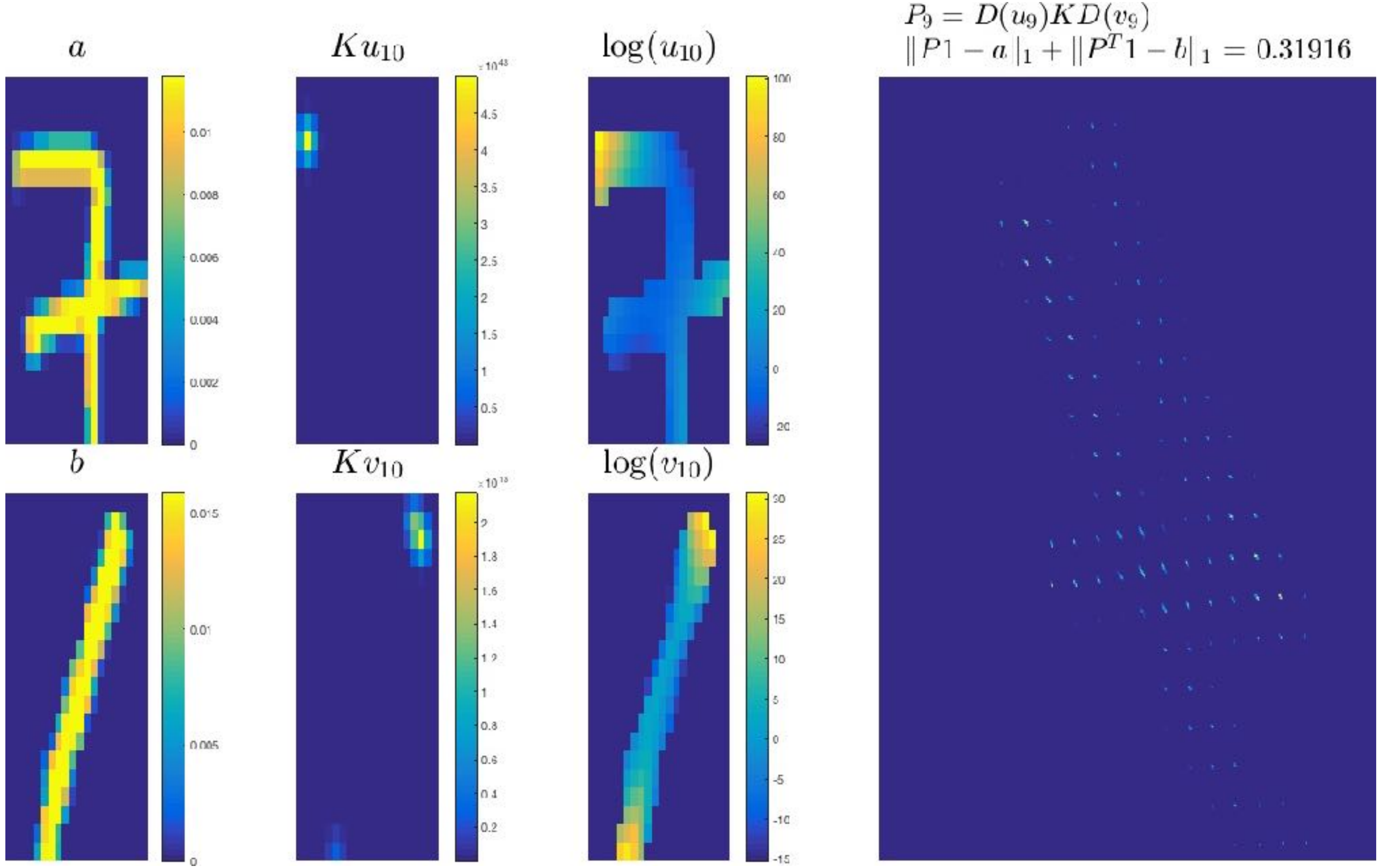
Very Fast EMD Approx. Solver



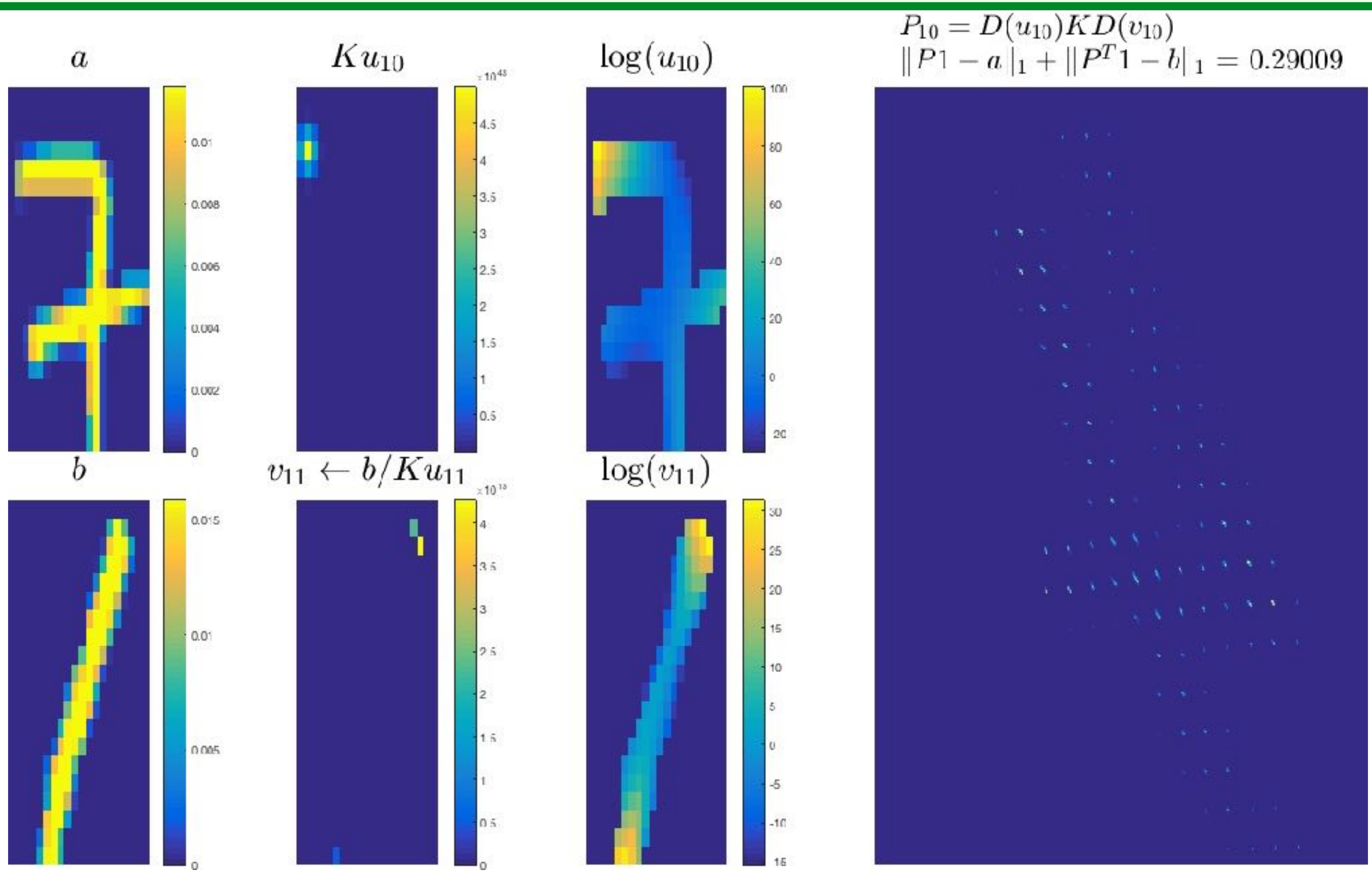
Very Fast EMD Approx. Solver



Very Fast EMD Approx. Solver



Very Fast EMD Approx. Solver



Sinkhorn as a Dual Algorithm

Def. Regularized Wasserstein, $\gamma \geq 0$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \min_{\boldsymbol{P} \in U(\boldsymbol{a}, \boldsymbol{b})} \langle \boldsymbol{P}, M_{\mathbf{XY}} \rangle - \gamma E(\boldsymbol{P})$$

REGULARIZED DISCRETE PRIMAL

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) = \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \boldsymbol{\alpha}^T \boldsymbol{a} + \boldsymbol{\beta}^T \boldsymbol{b} - \gamma (e^{\boldsymbol{\alpha}/\gamma})^T K (e^{\boldsymbol{\beta}/\gamma})$$

$$\text{where} \quad K = \left[e^{-\frac{\boldsymbol{D}^p(\boldsymbol{x_i}, \boldsymbol{y_j})}{\gamma}} \right]_{ij}$$

REGULARIZED DISCRETE DUAL

Sinkhorn = *Block Coordinate Ascent* on Dual

Block Coordinate Ascent, *a.k.a Sinkhorn*

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) = \max_{\boldsymbol{\alpha}, \boldsymbol{\beta}} \boldsymbol{\alpha}^T \boldsymbol{a} + \boldsymbol{\beta}^T \boldsymbol{b} - \gamma (e^{\boldsymbol{\alpha}/\gamma})^T \; K (e^{\boldsymbol{\beta}/\gamma})$$

REGULARIZED DISCRETE DUAL

$$\mathcal{E}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \boldsymbol{\alpha}^T \boldsymbol{a} + \boldsymbol{\beta}^T \boldsymbol{b} - \gamma (e^{\boldsymbol{\alpha}/\gamma})^T \; K e^{\boldsymbol{\beta}/\gamma}$$

Block Coordinate Ascent, *a.k.a Sinkhorn*

$$W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \boldsymbol{\alpha}^T \boldsymbol{a} + \boldsymbol{\beta}^T \boldsymbol{b} - \gamma(e^{\boldsymbol{\alpha}/\gamma})^T \; K(e^{\boldsymbol{\beta}/\gamma})$$

REGULARIZED DISCRETE DUAL

$$\mathcal{E}(\boldsymbol{\alpha},\boldsymbol{\beta}) = \boldsymbol{\alpha}^T \boldsymbol{a} + \boldsymbol{\beta}^T \boldsymbol{b} - \gamma(e^{\boldsymbol{\alpha}/\gamma})^T \; K e^{\boldsymbol{\beta}/\gamma}$$

$$\nabla_{\boldsymbol{\alpha}}\mathcal{E} = \boldsymbol{a} - e^{\boldsymbol{\alpha}/\gamma} \odot \; K e^{\boldsymbol{\beta}/\gamma}$$

$$\nabla_{\boldsymbol{\beta}}\mathcal{E} = \boldsymbol{b} - e^{\boldsymbol{\beta}/\gamma} \odot \; K^T e^{\boldsymbol{\alpha}/\gamma}$$

Block Coordinate Ascent, *a.k.a Sinkhorn*

$$W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \boldsymbol{\alpha}^T\boldsymbol{a} + \boldsymbol{\beta}^T\boldsymbol{b} - \gamma(e^{\boldsymbol{\alpha}/\gamma})^T\ K(e^{\boldsymbol{\beta}/\gamma})$$

REGULARIZED DISCRETE DUAL

$$\mathcal{E}(\boldsymbol{\alpha},\boldsymbol{\beta}) = \boldsymbol{\alpha}^T\boldsymbol{a} + \boldsymbol{\beta}^T\boldsymbol{b} - \gamma(e^{\boldsymbol{\alpha}/\gamma})^T\ K e^{\boldsymbol{\beta}/\gamma}$$

$$\nabla_{\boldsymbol{\alpha}}\mathcal{E} = \boldsymbol{a} - e^{\boldsymbol{\alpha}/\gamma} \odot\ K e^{\boldsymbol{\beta}/\gamma}$$

$$\boldsymbol{\alpha} \leftarrow \gamma\left(\log \boldsymbol{a} - \log\ K(e^{\boldsymbol{\beta}/\gamma})\right)$$

$$\nabla_{\boldsymbol{\beta}}\mathcal{E} = \boldsymbol{b} - e^{\boldsymbol{\beta}/\gamma} \odot\ K^T e^{\boldsymbol{\alpha}/\gamma}$$

$$\boldsymbol{\beta} \leftarrow \gamma\left(\log \boldsymbol{b} - \log\ K^T(e^{\boldsymbol{\alpha}/\gamma})\right)$$

Block Coordinate Ascent, *a.k.a Sinkhorn*

$$W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \boldsymbol{\alpha}^T\boldsymbol{a} + \boldsymbol{\beta}^T\boldsymbol{b} - \gamma(e^{\boldsymbol{\alpha}/\gamma})^T \; K(e^{\boldsymbol{\beta}/\gamma})$$

REGULARIZED DISCRETE DUAL

Block Coordinate Ascent, *a.k.a Sinkhorn*

$$W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \boldsymbol{\alpha}^T\boldsymbol{a} + \boldsymbol{\beta}^T\boldsymbol{b} - \gamma(e^{\boldsymbol{\alpha}/\gamma})^T\ K(e^{\boldsymbol{\beta}/\gamma})$$

REGULARIZED DISCRETE DUAL

$$(\boldsymbol{u},\boldsymbol{v}) \stackrel{\text{def}}{=} (e^{\boldsymbol{\alpha}/\gamma}, e^{\boldsymbol{\beta}/\gamma})$$

$$\boldsymbol{\alpha} \leftarrow \gamma\left(\log \boldsymbol{a} - \log\ K(e^{\boldsymbol{\beta}/\gamma})\right)$$

$$\boldsymbol{u} \leftarrow \frac{\boldsymbol{a}}{K\boldsymbol{v}}$$

$$\boldsymbol{\beta} \leftarrow \gamma\left(\log \boldsymbol{b} - \log\ K^T(e^{\boldsymbol{\alpha}/\gamma})\right)$$

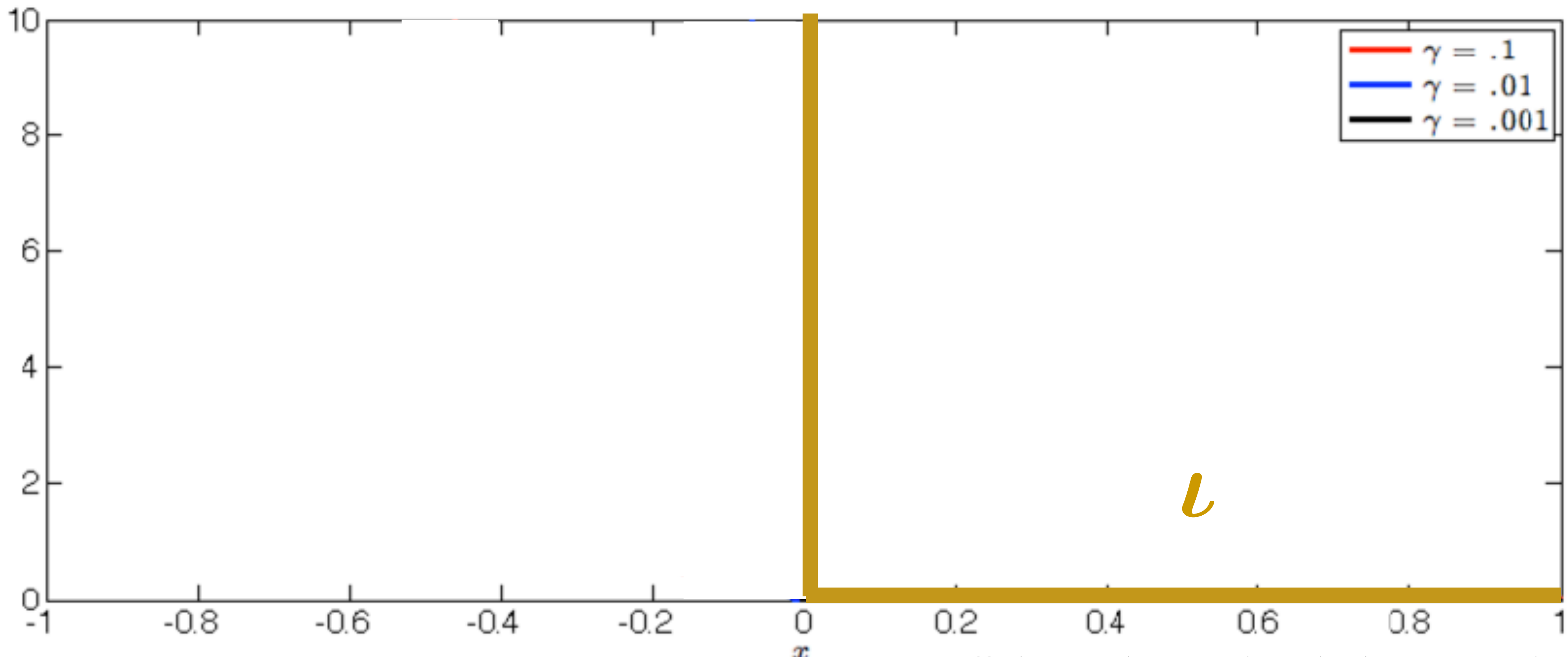
$$\boldsymbol{v} \leftarrow \frac{\boldsymbol{b}}{K^T\boldsymbol{u}}$$

Stochastic Formulation

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\varphi, \psi} \int \varphi d\boldsymbol{\mu} + \int \psi d\boldsymbol{\nu} - \iota_C(\varphi, \psi)$$

$$C = \{(\varphi, \psi) | \forall x, y, \varphi(x) + \psi(y) \leq \boldsymbol{D}(x, y)^p\}$$

DUAL



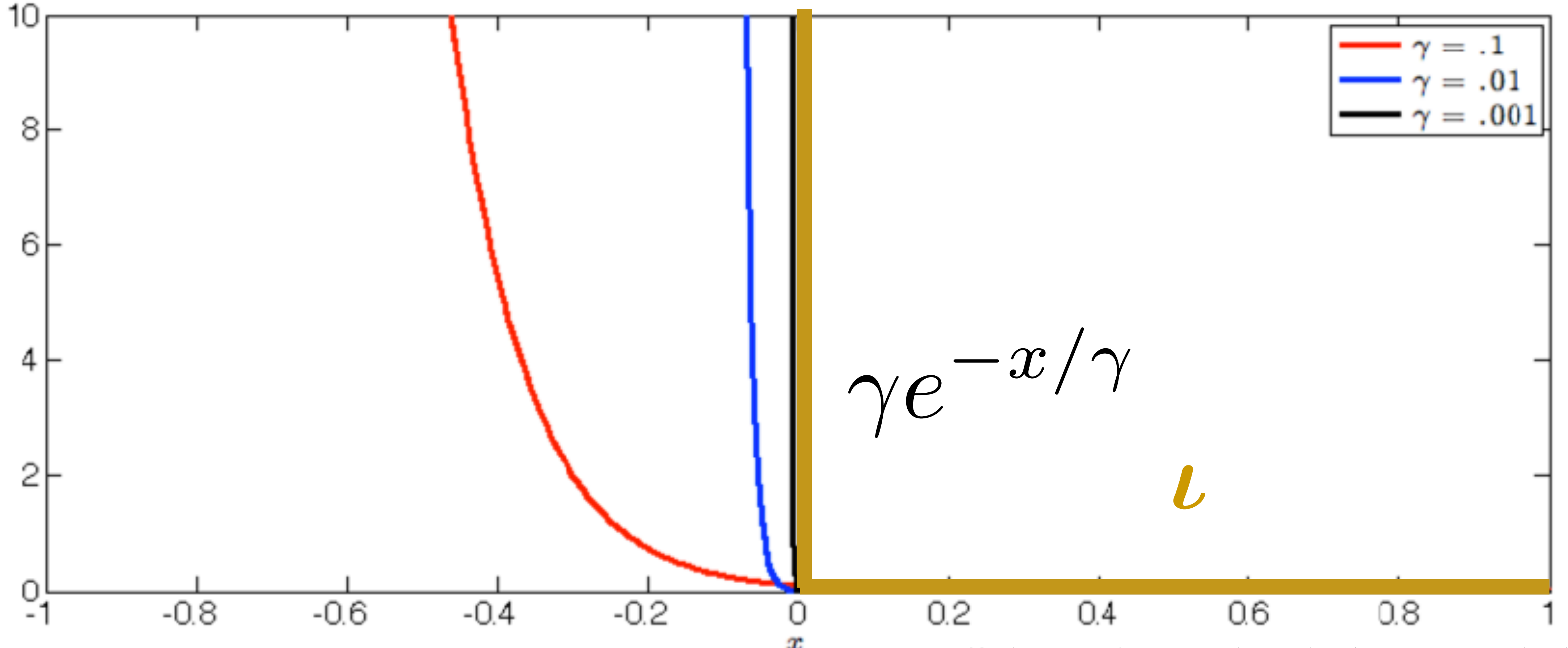
$\boldsymbol{D}^p(x, y) - (\varphi(x) + \psi(y))$

Stochastic Formulation

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iota_C(\boldsymbol{\varphi},\boldsymbol{\psi})$$

$$C = \{(\boldsymbol{\varphi},\boldsymbol{\psi}) | \forall x, y, \boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y) \leq \boldsymbol{D}(x,y)^p\}$$

DUAL



$\boldsymbol{D}^p(x,y) - (\boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y))$

Stochastic Formulation

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iota_C(\boldsymbol{\varphi},\boldsymbol{\psi})$$

$$C = \{(\boldsymbol{\varphi},\boldsymbol{\psi}) | \forall x, y, \boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y) \leq \boldsymbol{D}(x,y)^p\}$$

DUAL

regularizing dual *constraints* $\gamma > 0$

$$W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iota_C^\gamma(\boldsymbol{\varphi},\boldsymbol{\psi})$$

$$\iota_C^\gamma(\boldsymbol{\varphi},\boldsymbol{\psi}) = \gamma \iint e^{(\boldsymbol{\varphi} \oplus \boldsymbol{\psi} - \boldsymbol{D}^p)/\gamma} d\boldsymbol{\mu} d\boldsymbol{\nu}$$

REGULARIZED DUAL

Stochastic Formulation

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iota_C(\boldsymbol{\varphi},\boldsymbol{\psi})$$

$$C = \{(\boldsymbol{\varphi},\boldsymbol{\psi}) | \forall x, y, \boldsymbol{\varphi}(x) + \boldsymbol{\psi}(y) \leq \boldsymbol{D}(x,y)^p\}$$

DUAL

regularizing dual *constraints* $\gamma > 0$

$$W_{\gamma}(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi},\boldsymbol{\psi}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\psi} d\boldsymbol{\nu} - \iota_C^{\gamma}(\boldsymbol{\varphi},\boldsymbol{\psi})$$

$$\iota_C^{\gamma}(\boldsymbol{\varphi},\boldsymbol{\psi}) = \gamma \iint e^{(\boldsymbol{\varphi} \oplus \boldsymbol{\psi} - \boldsymbol{D}^p)/\gamma} d\boldsymbol{\mu} d\boldsymbol{\nu}$$

REGULARIZED DUAL

Smoothed D transforms

$$W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\varphi}^{\boldsymbol{D}} d\boldsymbol{\nu}.$$

SEMI-DUAL

$\gamma > 0$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi}} \int \boldsymbol{\varphi} d\boldsymbol{\mu} + \int \boldsymbol{\varphi}^{\boldsymbol{D},\gamma} d\boldsymbol{\nu}.$$

$$\boldsymbol{\varphi}^{\boldsymbol{D},\gamma} = -\gamma \log \int e^{\frac{\boldsymbol{\varphi}(x) - \boldsymbol{D}(x,\cdot)^p}{\gamma}} d\boldsymbol{\mu}(x)$$

REGULARIZED SEMI-DUAL

Regularized Semidual Wasserstein

$$W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = \sup_{\boldsymbol{\varphi}} \int \boldsymbol{\varphi}\, d\boldsymbol{\mu} + \int \boldsymbol{\varphi}^{\boldsymbol{D},\gamma} d\boldsymbol{\nu}.$$

$$\boldsymbol{\varphi}^{\boldsymbol{D},\gamma} = -\gamma \log \int e^{\frac{\boldsymbol{\varphi}(x) - \boldsymbol{D}(x,\cdot)^p}{\gamma}} d\boldsymbol{\mu}(x)$$

REGULARIZED SEMI-DUAL

substituting

$$\sup_{\boldsymbol{\varphi}} \int_y \left[\int_x \boldsymbol{\varphi}(x) d\boldsymbol{\mu}(x) - \gamma \log \int_x e^{\frac{\boldsymbol{\varphi}(x) - \boldsymbol{D}(x,y)^p}{\gamma}} d\boldsymbol{\mu}(x) \right] d\boldsymbol{\nu}(y).$$

REGULARIZED SEMI-DUAL

Stochastic Regularized Semidual

$$\sup_{\boldsymbol{\varphi}} \int_y \left[\int_x \boldsymbol{\varphi}(x) d\boldsymbol{\mu}(x) - \gamma \log \int_x e^{\frac{\boldsymbol{\varphi}(x) - \boldsymbol{D}(x,y)^p}{\gamma}} d\boldsymbol{\mu}(x) \right] d\boldsymbol{\nu}(y).$$

REGULARIZED SEMI-DUAL

Stochastic Regularized Semidual

$$\sup_{\boldsymbol{\varphi}} \int_y \left[\int_x \boldsymbol{\varphi}(x) d\boldsymbol{\mu}(x) - \gamma \log \int_x e^{\frac{\boldsymbol{\varphi}(x) - \boldsymbol{D}(x,y)^p}{\gamma}} d\boldsymbol{\mu}(x) \right] d\boldsymbol{\nu}(y).$$

REGULARIZED SEMI-DUAL

What if $\boldsymbol{\mu}$ is a discrete measure?

$\boldsymbol{\mu} = \sum_{i=1}^n \boldsymbol{a_i} \delta_{\boldsymbol{x_i}}$

$\boldsymbol{\varphi} \in L_1(\boldsymbol{\mu})$ is now just a vector $\boldsymbol{\alpha} \in \mathbb{R}^n$!

Stochastic Regularized Semidual

$$\sup_{\varphi} \int_y \left[\int_x \varphi(x) d\mu(x) - \gamma \log \int_x e^{\frac{\varphi(x) - D(x,y)^p}{\gamma}} d\mu(x) \right] d\nu(y).$$

REGULARIZED SEMI-DUAL

What if μ is a discrete measure?

$\mu = \sum_{i=1}^n a_i \delta_{x_i}$

$\varphi \in L_1(\mu)$ is now just a vector $\alpha \in \mathbb{R}^n$!

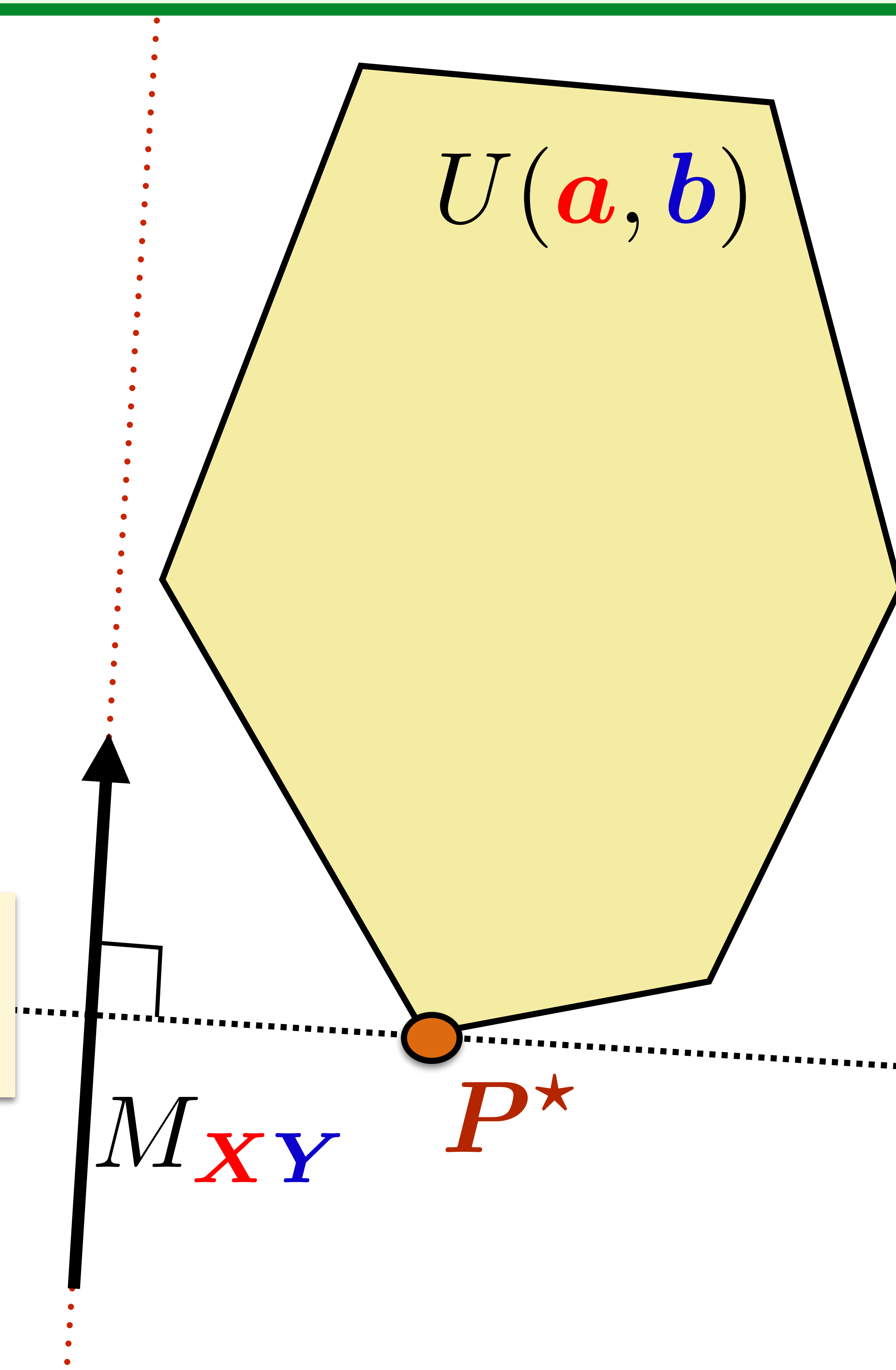
$$\sup_{\alpha \in \mathbb{R}^n} \int_y \left[\sum_{i=1}^n \alpha_i a_i - \gamma \log \sum_{i=1}^n e^{\frac{\alpha_i - D(x_i, y)^p}{\gamma}} a_i \right] d\nu(y)$$

$$= \sup_{\alpha \in \mathbb{R}^n} \mathbb{E}_{\nu}[f(\alpha, y)]$$

STOCHASTIC REGULARIZED SEMI-DUAL

Sinkhorn in between W and MMD

$$\mu = \sum_{i=1}^{n} a_i \delta_{x_i} \qquad \nu = \sum_{j=1}^{m} b_j \delta_{y_j}$$



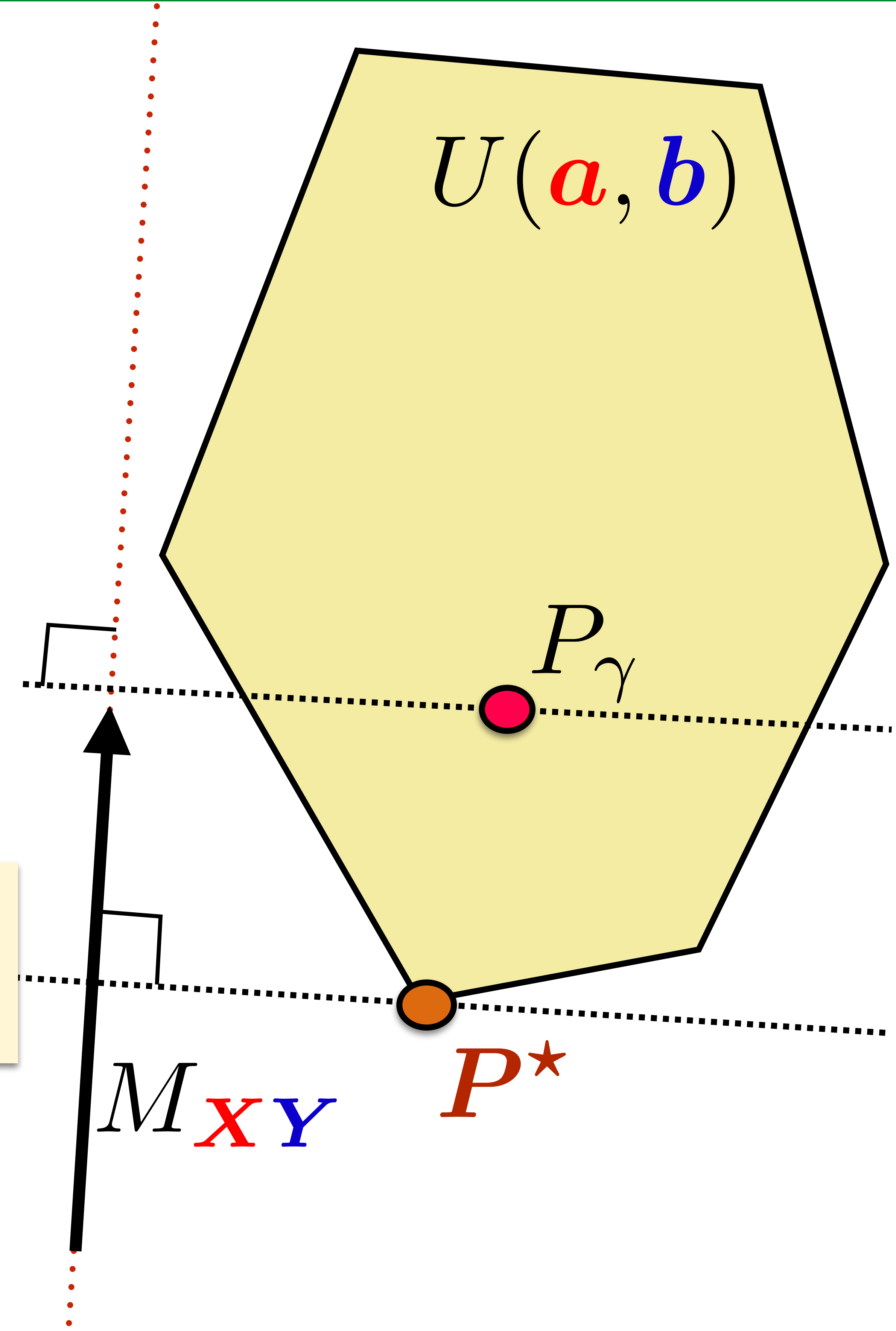
$$W^p(\mu, \nu) = \langle P^\star, M_{XY} \rangle$$

Sinkhorn in between W and MMD

$$\boldsymbol{\mu} = \sum_{i=1}^{n} \boldsymbol{a}_i \delta_{\boldsymbol{x}_i} \qquad \boldsymbol{\nu} = \sum_{j=1}^{m} \boldsymbol{b}_j \delta_{\boldsymbol{y}_j}$$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle P_\gamma, M_{\mathbf{XY}} \rangle$$

$$W^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle \boldsymbol{P}^\star, M_{\mathbf{XY}} \rangle$$



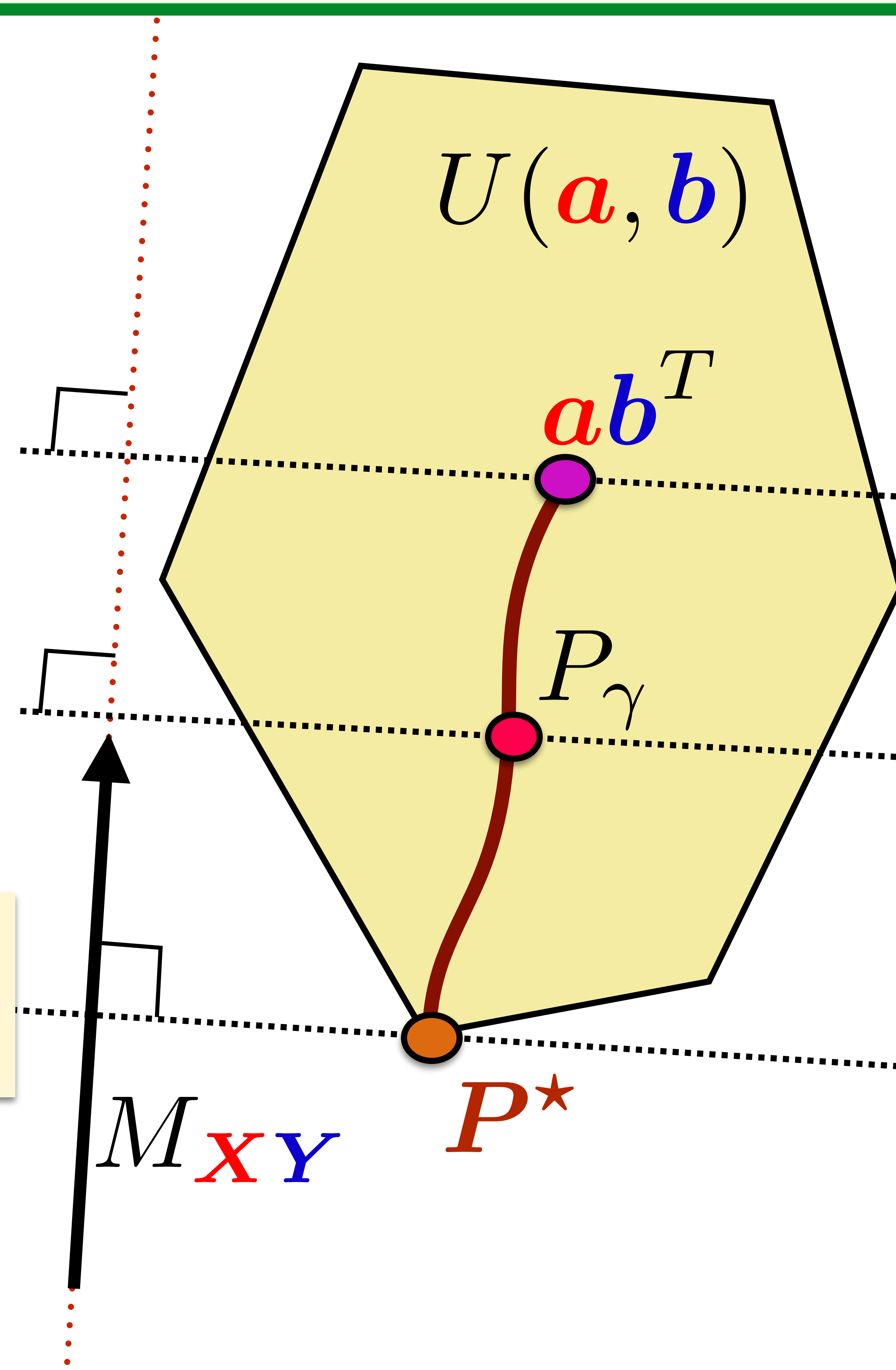
Sinkhorn in between W and MMD

$$\boldsymbol{\mu} = \sum_{i=1}^{n} a_i \delta_{x_i} \qquad \boldsymbol{\nu} = \sum_{j=1}^{m} b_j \delta_{y_j}$$

$$\mathcal{E}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle \boldsymbol{a}\boldsymbol{b}^T, M_{\mathbf{XY}} \rangle$$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle P_\gamma, M_{\mathbf{XY}} \rangle$$

$$W^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle \boldsymbol{P}^\star, M_{\mathbf{XY}} \rangle$$



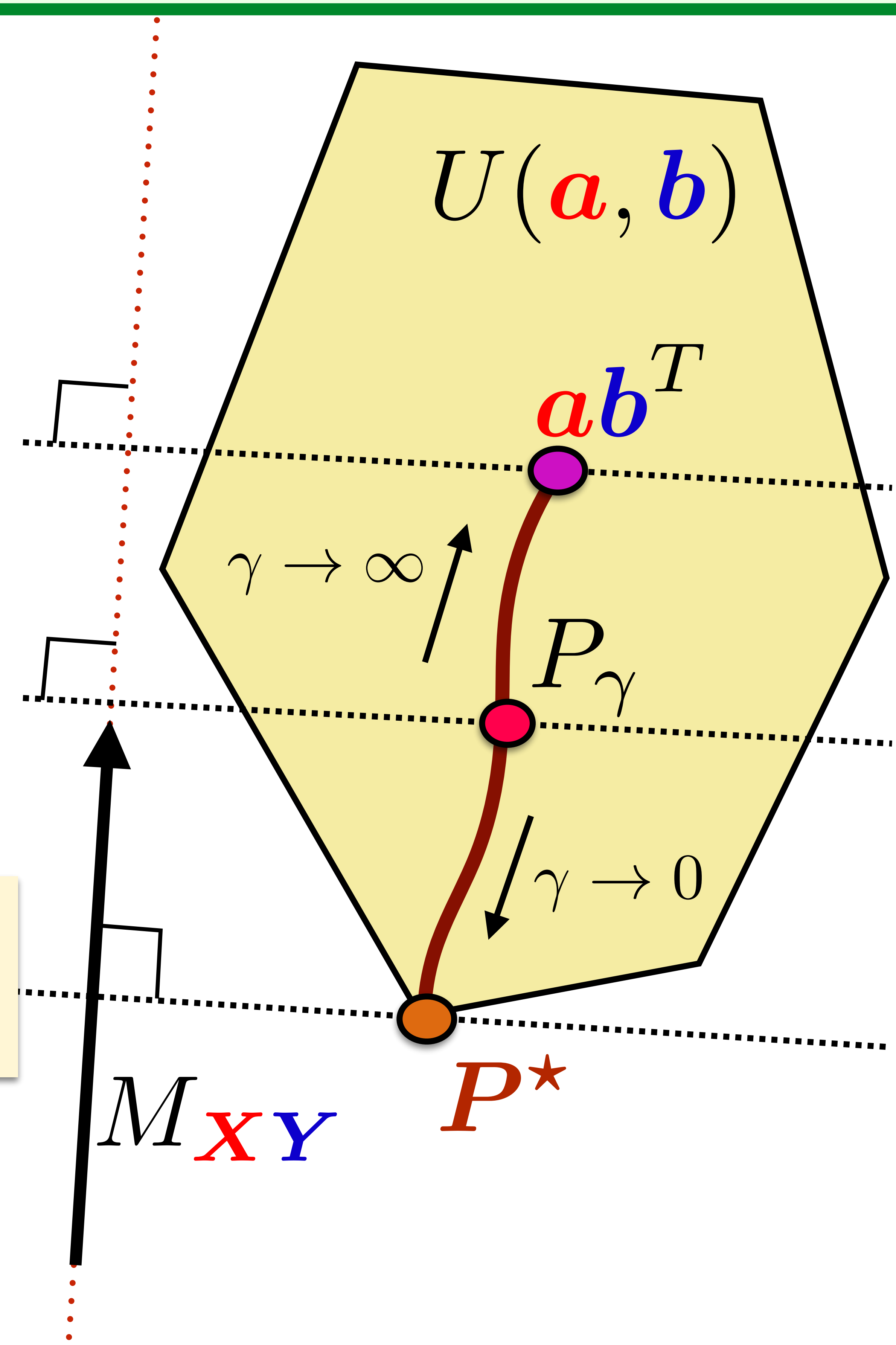
Sinkhorn in between W and MMD

$$\boldsymbol{\mu} = \sum_{i=1}^{n} \boldsymbol{a}_i \boldsymbol{\delta}_{\boldsymbol{x}_i} \qquad \boldsymbol{\nu} = \sum_{j=1}^{m} \boldsymbol{b}_j \boldsymbol{\delta}_{\boldsymbol{y}_j}$$

$$\mathcal{E}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle \boldsymbol{a}\boldsymbol{b}^T, M_{\mathbf{XY}} \rangle$$

$$W_\gamma(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle P_\gamma, M_{\mathbf{XY}} \rangle$$

$$W^p(\boldsymbol{\mu}, \boldsymbol{\nu}) = \langle \boldsymbol{P}^\star, M_{\mathbf{XY}} \rangle$$



Sinkhorn in between W and MMD

$$\mathcal{E}(\boldsymbol{\mu},\boldsymbol{\nu}) = \langle \boldsymbol{a}\boldsymbol{b}^T, M_{\mathbf{X}\mathbf{Y}} \rangle$$

$$\mathcal{MMD}(\boldsymbol{\mu},\boldsymbol{\nu}) = \mathcal{E}(\boldsymbol{\mu},\boldsymbol{\nu}) - \frac{1}{2}(\mathcal{E}(\boldsymbol{\mu},\boldsymbol{\mu}) + \mathcal{E}(\boldsymbol{\nu},\boldsymbol{\nu}))$$

$$W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = \langle P_\gamma, M_{\mathbf{X}\mathbf{Y}} \rangle$$

$$\bar{W}_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) - \frac{1}{2}(W_\gamma(\boldsymbol{\mu},\boldsymbol{\mu}) + W_\gamma(\boldsymbol{\nu},\boldsymbol{\nu}))$$

$$W^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \langle \boldsymbol{P}^\star, M_{\mathbf{X}\mathbf{Y}} \rangle$$

Sinkhorn in between W and MMD

$$\mathcal{MMD}(\boldsymbol{\mu},\boldsymbol{\nu}) = \mathcal{E}(\boldsymbol{\mu},\boldsymbol{\nu}) - \frac{1}{2}(\mathcal{E}(\boldsymbol{\mu},\boldsymbol{\mu}) + \mathcal{E}(\boldsymbol{\nu},\boldsymbol{\nu}))$$

$$\gamma \to \infty \uparrow$$

$$\bar{W}_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) - \frac{1}{2}(W_\gamma(\boldsymbol{\mu},\boldsymbol{\mu}) + W_\gamma(\boldsymbol{\nu},\boldsymbol{\nu}))$$

$$\gamma \to 0 \downarrow$$

$$W^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \langle \boldsymbol{P}^\star, M_{\boldsymbol{XY}} \rangle$$

How to compare them?

i.i.d samples $\boldsymbol{x_1, \dots, x_n} \sim \boldsymbol{\mu}$, $\boldsymbol{y_1, \dots, y_m} \sim \boldsymbol{\nu}$,

$$\hat{\boldsymbol{\mu}}_{\boldsymbol{n}} \stackrel{\text{def}}{=} \frac{1}{n} \sum_i \delta_{\boldsymbol{x_i}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}} \stackrel{\text{def}}{=} \frac{1}{m} \sum_j \delta_{\boldsymbol{y_j}}$$

Computational properties

Effort to compute/approximate $\boldsymbol{\Delta}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{m}})$?

Statistical properties

$$|\boldsymbol{\Delta}(\boldsymbol{\mu}, \boldsymbol{\nu}) - \boldsymbol{\Delta}(\hat{\boldsymbol{\mu}}_{\boldsymbol{n}}, \hat{\boldsymbol{\nu}}_{\boldsymbol{n}})| \leq f(n)?$$

Sinkhorn in between W and MMD

$$\mathcal{MMD}(\boldsymbol{\mu},\boldsymbol{\nu}) = \mathcal{E}(\boldsymbol{\mu},\boldsymbol{\nu}) - \frac{1}{2}(\mathcal{E}(\boldsymbol{\mu},\boldsymbol{\mu}) + \mathcal{E}(\boldsymbol{\nu},\boldsymbol{\nu}))$$

$$\boxed{(n+m)^2} \qquad \boxed{O(1/\sqrt{n})}$$

$$W^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \langle \boldsymbol{P}^\star, M_{\mathbf{XY}} \rangle$$

$$\boxed{O((n+m)nm\log(n+m))} \qquad \boxed{O(1/n^{1/d})}$$

Sinkhorn in between W and MMD

$$\mathcal{MMD}(\boldsymbol{\mu},\boldsymbol{\nu}) = \mathcal{E}(\boldsymbol{\mu},\boldsymbol{\nu}) - \frac{1}{2}(\mathcal{E}(\boldsymbol{\mu},\boldsymbol{\mu}) + \mathcal{E}(\boldsymbol{\nu},\boldsymbol{\nu}))$$

$(n+m)^2$

$O(1/\sqrt{n})$

$$\bar{W}_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) = W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu}) - \frac{1}{2}(W_\gamma(\boldsymbol{\mu},\boldsymbol{\mu}) + W_\gamma(\boldsymbol{\nu},\boldsymbol{\nu}))$$

$O((n+m)^2)$

$O\left(\frac{1}{\gamma^{d/2}\sqrt{n}}\right)$

[GCBCP'18]

[FSVATP'18]

$$W^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \langle \boldsymbol{P}^\star, M_{\mathbf{XY}} \rangle$$

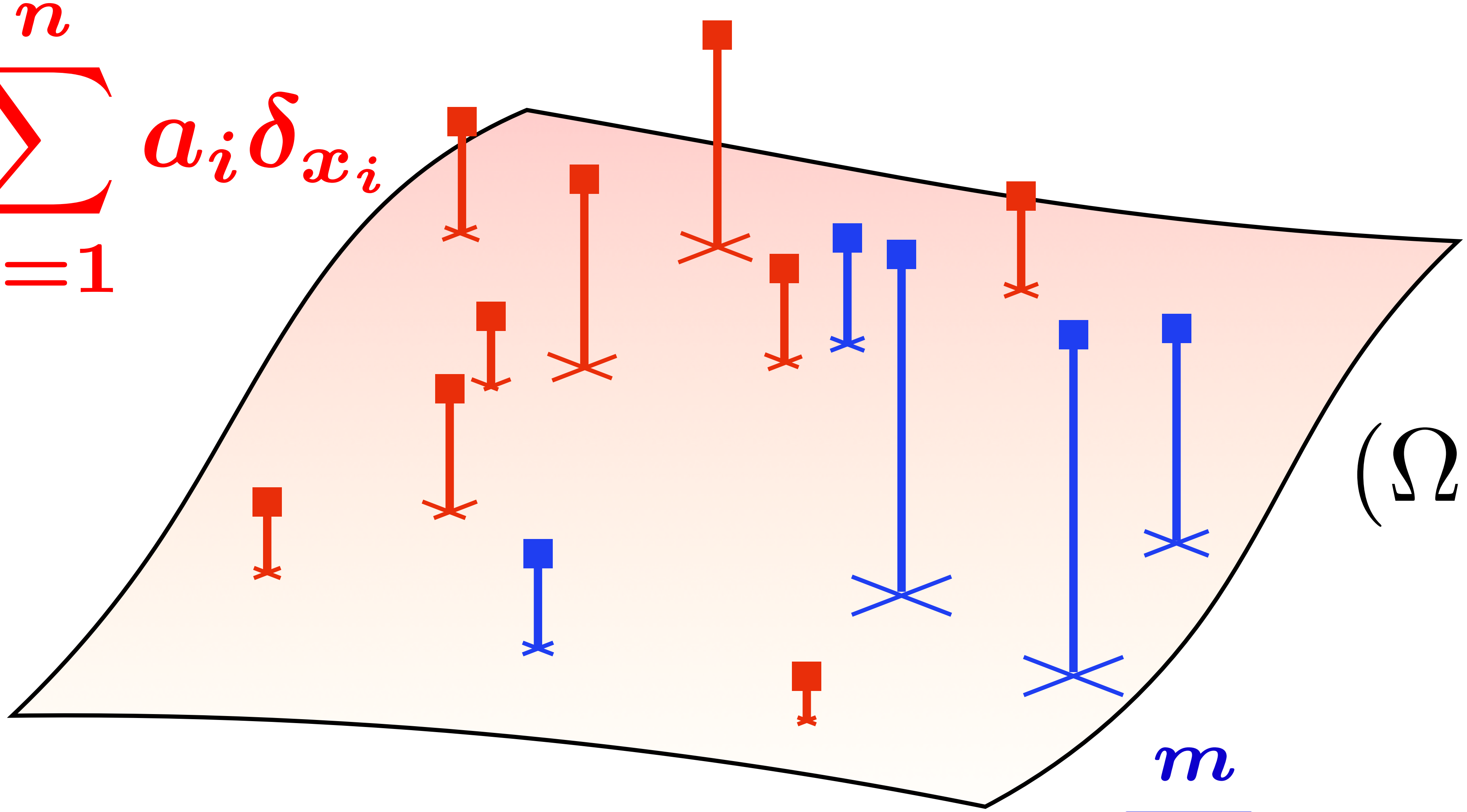
$O((n+m)nm\log(n+m))$

$O(1/n^{1/d})$

Differentiability of W

$$W((\boldsymbol{a}, \boldsymbol{X}), (\boldsymbol{b}, \boldsymbol{Y}))$$

$$\boldsymbol{\mu} = \sum_{i=1}^{n} \boldsymbol{a_i} \boldsymbol{\delta_{x_i}}$$



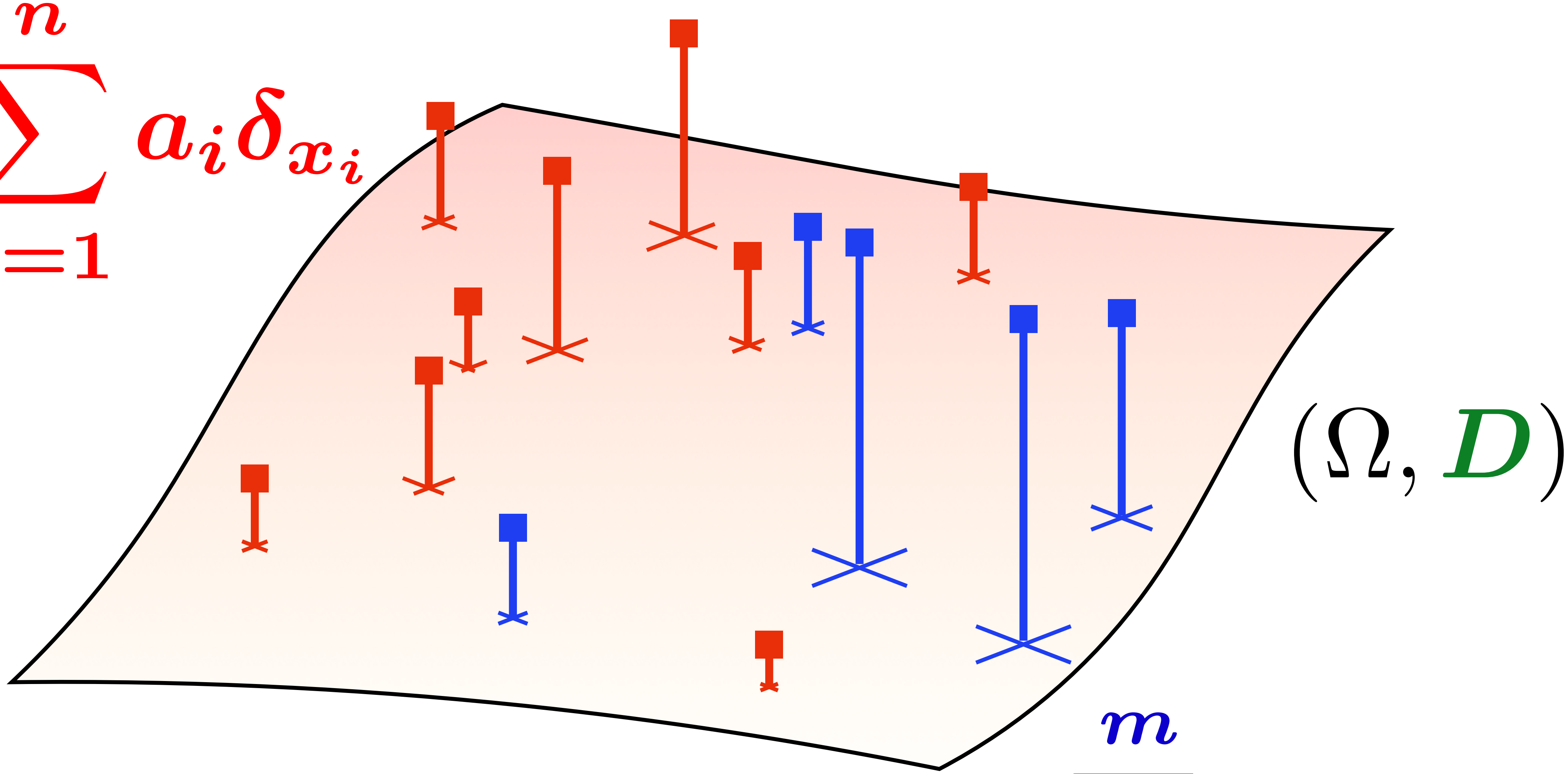
$$(\Omega, \boldsymbol{D})$$

$$\boldsymbol{\nu} = \sum_{j=1}^{m} \boldsymbol{b_j} \boldsymbol{\delta_{y_j}}$$

Differentiability of W

$$W((\boldsymbol{a} + \boldsymbol{\Delta a}, \boldsymbol{X}), (\boldsymbol{b}, \boldsymbol{Y})) = W((\boldsymbol{a}, \boldsymbol{X}), (\boldsymbol{b}, \boldsymbol{Y})) + ??$$

$$\boldsymbol{\mu} = \sum_{i=1}^{n} a_i \delta_{x_i}$$

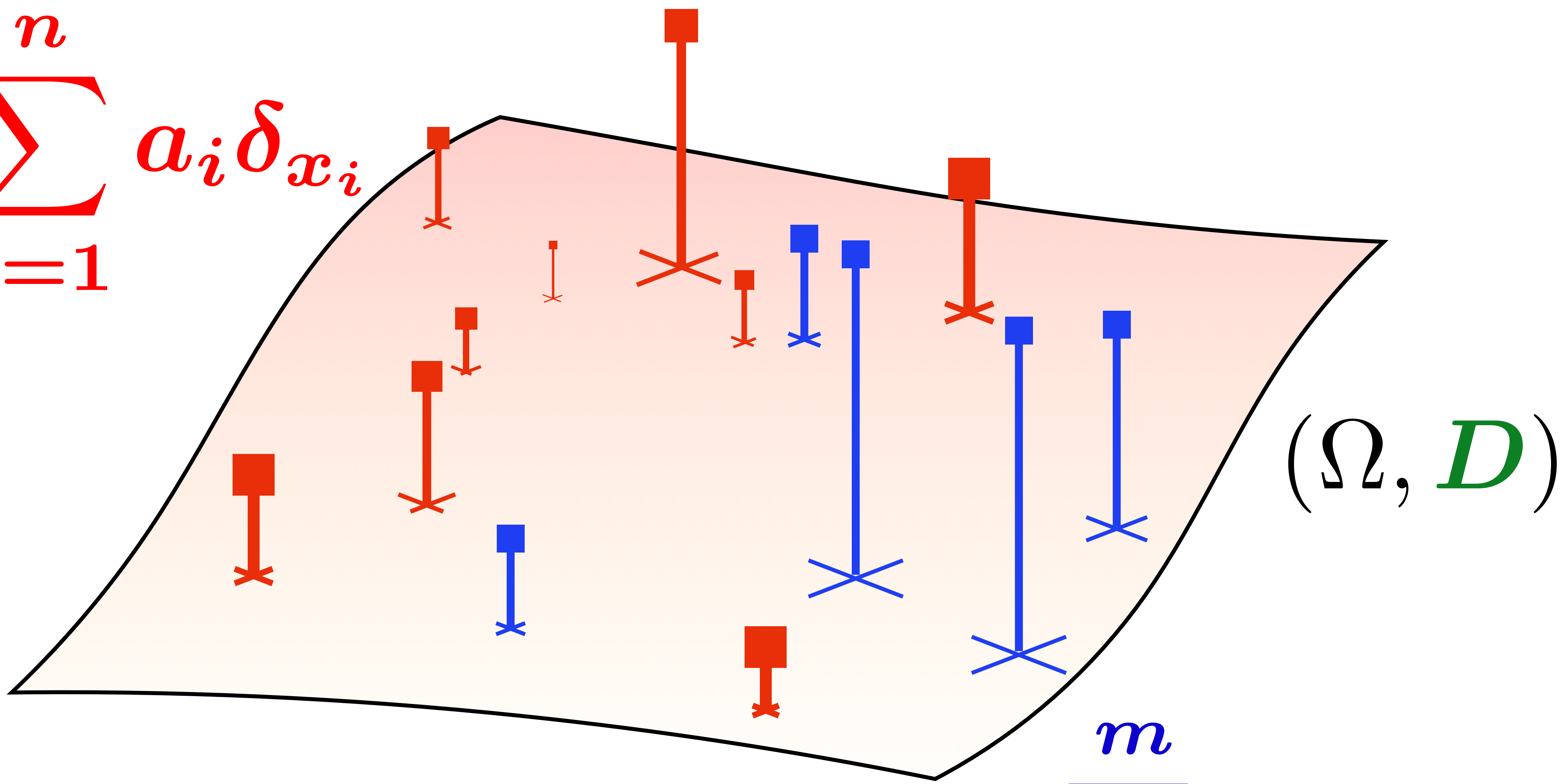


$$\boldsymbol{\nu} = \sum_{j=1}^{m} b_j \delta_{y_j}$$

Differentiability of W

$$W((\boldsymbol{a}+\boldsymbol{\Delta a},\boldsymbol{X}),(\boldsymbol{b},\boldsymbol{Y}))=W((\boldsymbol{a},\boldsymbol{X}),(\boldsymbol{b},\boldsymbol{Y}))+??$$

$$\boldsymbol{\mu}=\sum_{i=1}^{n}\boldsymbol{a_i}\boldsymbol{\delta_{x_i}}$$



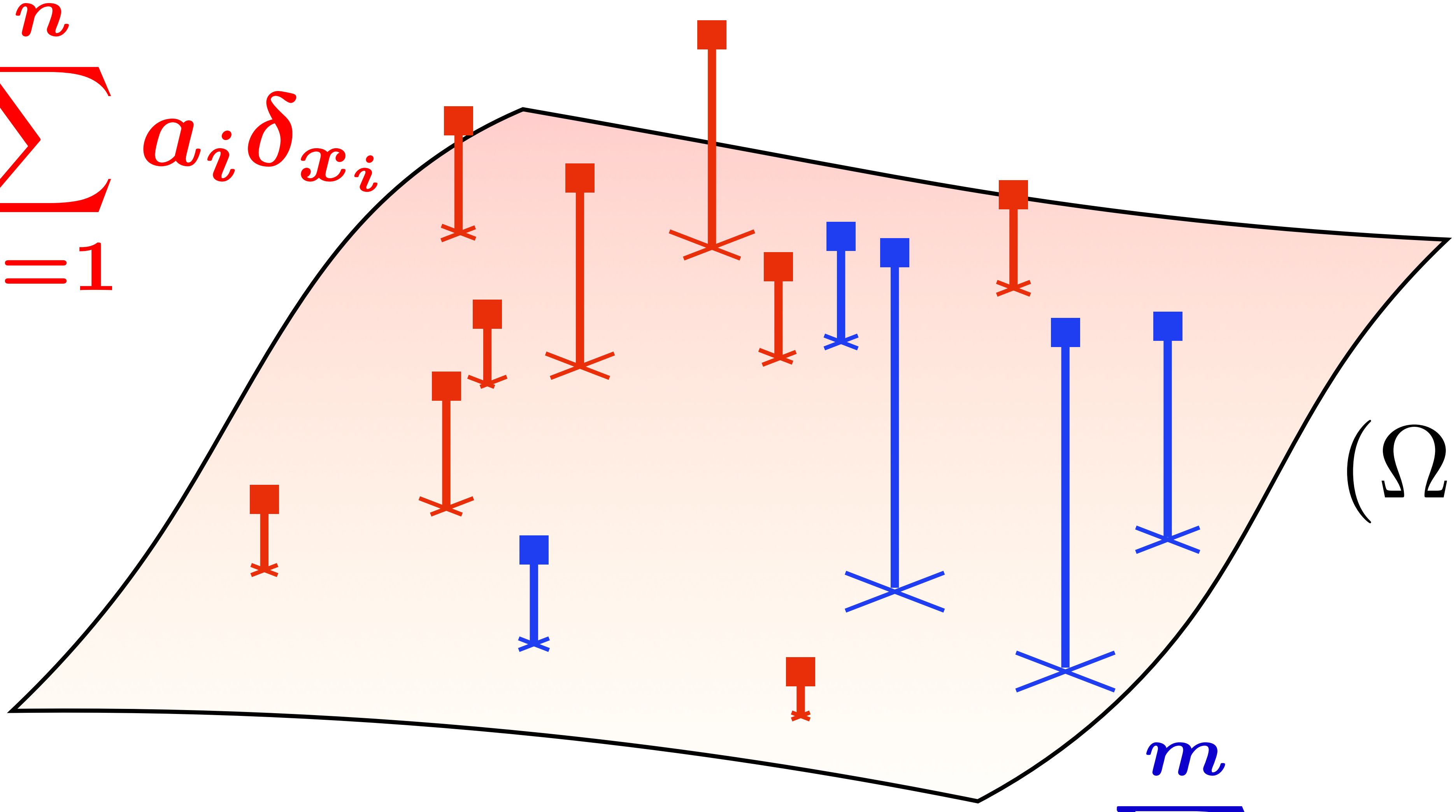
$$\boldsymbol{\nu}=\sum_{j=1}^{m}\boldsymbol{b_j}\boldsymbol{\delta_{y_j}}$$

$$\boldsymbol{a}\leftarrow\boldsymbol{a}+\boldsymbol{\Delta a}$$

Sinkhorn $\leadsto$ *Differentiability*

$$W((\boldsymbol{a},\boldsymbol{X}+\boldsymbol{\Delta X}),(\boldsymbol{b},\boldsymbol{Y})) = W((\boldsymbol{a},\boldsymbol{X}),(\boldsymbol{b},\boldsymbol{Y}))+??$$

$$\boldsymbol{\mu} = \sum_{i=1}^{n} \boldsymbol{a_i \delta_{x_i}}$$

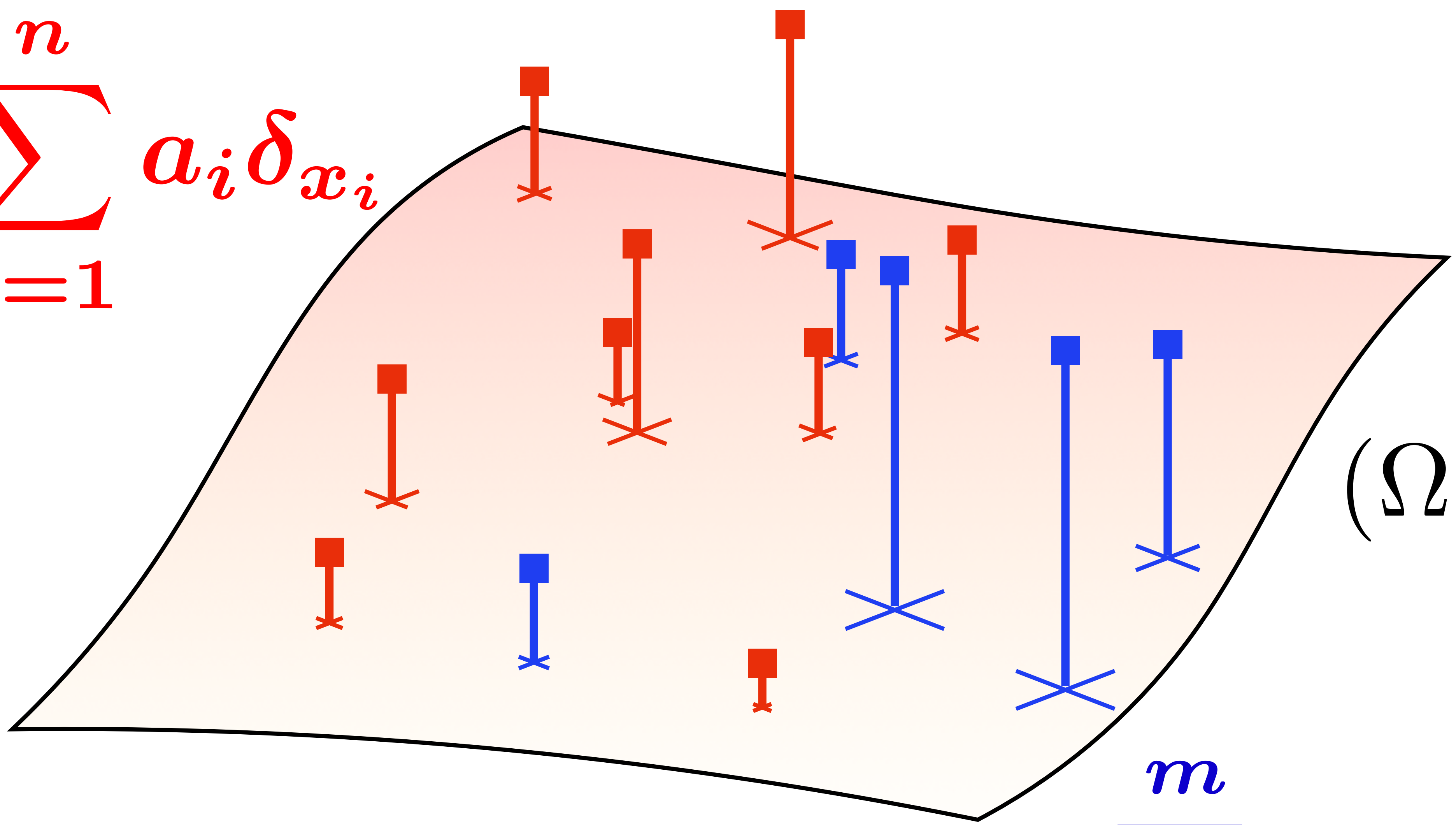


$$\boldsymbol{\nu} = \sum_{j=1}^{m} \boldsymbol{b_j \delta_{y_j}}$$

Sinkhorn ⇝ *Differentiability*

$$W((\boldsymbol{a}, \boldsymbol{X} + \boldsymbol{\Delta X}), (\boldsymbol{b}, \boldsymbol{Y})) = W((\boldsymbol{a}, \boldsymbol{X}), (\boldsymbol{b}, \boldsymbol{Y})) + ??$$

$$\boldsymbol{\mu} = \sum_{i=1}^{n} \boldsymbol{a_i \delta_{x_i}}$$



$$(\Omega, \boldsymbol{D})$$

$$\boldsymbol{X} \leftarrow \boldsymbol{X} + \boldsymbol{\Delta X}$$

$$\boldsymbol{\nu} = \sum_{j=1}^{m} \boldsymbol{b_j \delta_{y_j}}$$

How to decrease W? change weights

$$W_p^p(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\substack{\boldsymbol{\alpha}\in\mathbb{R}^n,\boldsymbol{\beta}\in\mathbb{R}^m\\ \boldsymbol{\alpha}\oplus\boldsymbol{\beta}\leq M_{\mathbf{X}\mathbf{Y}}}} \boldsymbol{\alpha}^T\boldsymbol{a} + \boldsymbol{\beta}^T\boldsymbol{b}.$$

DUAL

Prop. $W(\boldsymbol{\mu},\boldsymbol{\nu})$ is convex w.r.t. $\boldsymbol{a}$,

$$\partial_{\boldsymbol{a}} W = \arg_{\boldsymbol{\alpha}} \max_{\boldsymbol{\alpha}\oplus\boldsymbol{\beta}\leq M_{\mathbf{X}\mathbf{Y}}} \boldsymbol{\alpha}^T\boldsymbol{a} + \boldsymbol{\beta}^T\boldsymbol{b}.$$

Prop. $W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu})$ is convex and **differentiable** w.r.t. $\boldsymbol{a}$, $\nabla_{\boldsymbol{a}} W_\gamma = \boldsymbol{\alpha}_\gamma^\star = \gamma\log\boldsymbol{u}$

How to decrease W? change locations

$$W_{\mathbf{2}}^{\mathbf{2}}(\boldsymbol{\mu},\boldsymbol{\nu}) = \min_{\substack{\boldsymbol{P}\in\mathbb{R}_+^{n\times m}\\ \boldsymbol{P}\mathbf{1}_m=\boldsymbol{a},\boldsymbol{P}^T\mathbf{1}_n=\boldsymbol{b}}} \langle \boldsymbol{P}, \mathbf{1}_n\mathbf{1}_d^T\boldsymbol{X}^2 + \boldsymbol{Y}^{2T}\mathbf{1}_d\mathbf{1}_m - 2\boldsymbol{X}^T\boldsymbol{Y}\rangle$$

PRIMAL

Prop. $p=2, \Omega=\mathbb{R}^d$. $W(\boldsymbol{\mu},\boldsymbol{\nu})$ decreases if

$$\boldsymbol{X} \leftarrow \boldsymbol{Y}P^{\star T}\mathbf{D}(\boldsymbol{a}^{-1}).$$

Prop. $p=2, \Omega=\mathbb{R}^d$. $W_\gamma(\boldsymbol{\mu},\boldsymbol{\nu})$ is **differentiable** w.r.t. $\boldsymbol{X}$, with

$$\nabla_{\boldsymbol{X}}W_\gamma = \boldsymbol{X} - \boldsymbol{Y}P_\gamma^T\mathbf{D}(\boldsymbol{a}^{-1}).$$

Sinkhorn: A Programmer View

Def. For $L \geq 1$, define

$$W_L(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \langle \boldsymbol{P_L}, M_{\mathbf{XY}} \rangle,$$

where $\boldsymbol{P_L} \stackrel{\text{def}}{=} \mathbf{diag}(\boldsymbol{u_L}) K \mathbf{diag}(\boldsymbol{v_L}),$

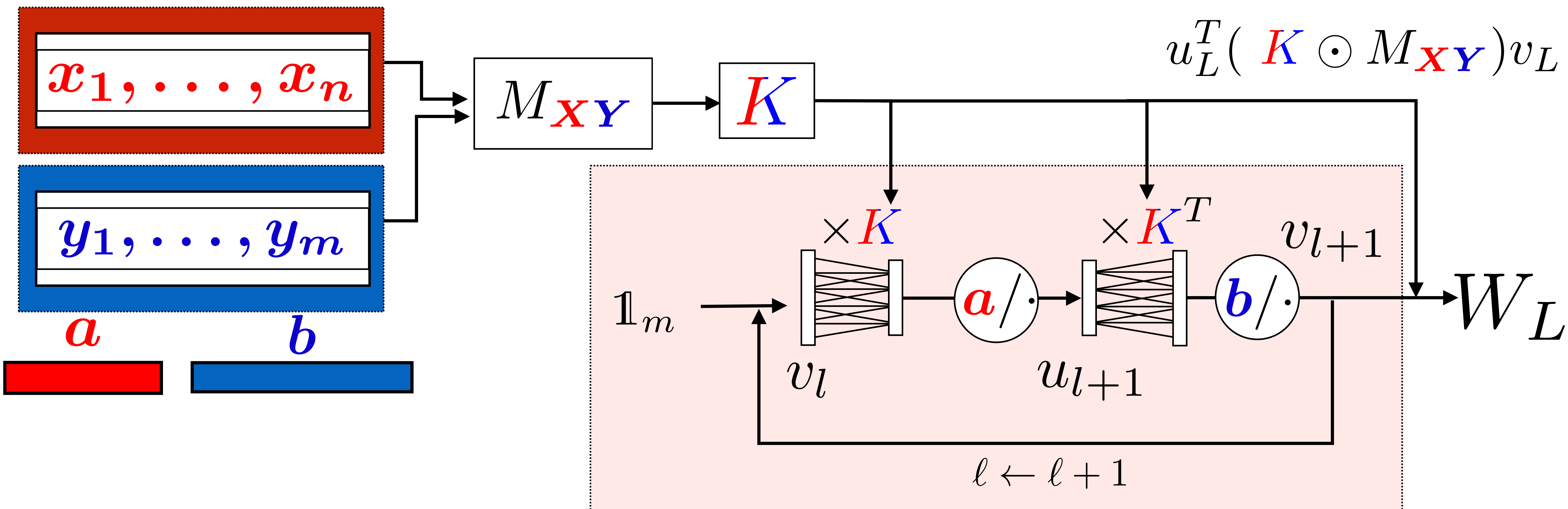
$\boldsymbol{v_0} = \mathbf{1}_m; l \geq 0, \boldsymbol{u_l} \stackrel{\text{def}}{=} \boldsymbol{a}/K\boldsymbol{v_l}, \boldsymbol{v_{l+1}} \stackrel{\text{def}}{=} \boldsymbol{b}/K^T\boldsymbol{u_l}.$

Prop. $\frac{\partial W_L}{\partial \mathbf{X}}, \frac{\partial W_L}{\partial \boldsymbol{a}}$ can be computed recursively, in $O(L)$ kernel $K \times$vector products.

Sinkhorn: A Programmer View

Def. For $L \geq 1$, define

$$W_L(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \langle \boldsymbol{P_L}, M_{\boldsymbol{XY}} \rangle,$$



Sinkhorn $\ell = 1, \dots, L-1$

Sinkhorn: A Programmer View

Def. For $L \geq 1$, define

$$W_L(\boldsymbol{\mu}, \boldsymbol{\nu}) \stackrel{\text{def}}{=} \langle \boldsymbol{P_L}, M_{\boldsymbol{XY}} \rangle,$$

Prop. $\frac{\partial W_L}{\partial \boldsymbol{X}}, \frac{\partial W_L}{\partial \boldsymbol{a}}$ can be computed recursively, in $O(L)$ kernel $K \times$vector products.

[Hashimoto'16] [Bonneel'16][Shalit'16]

Recently: projection as regularization

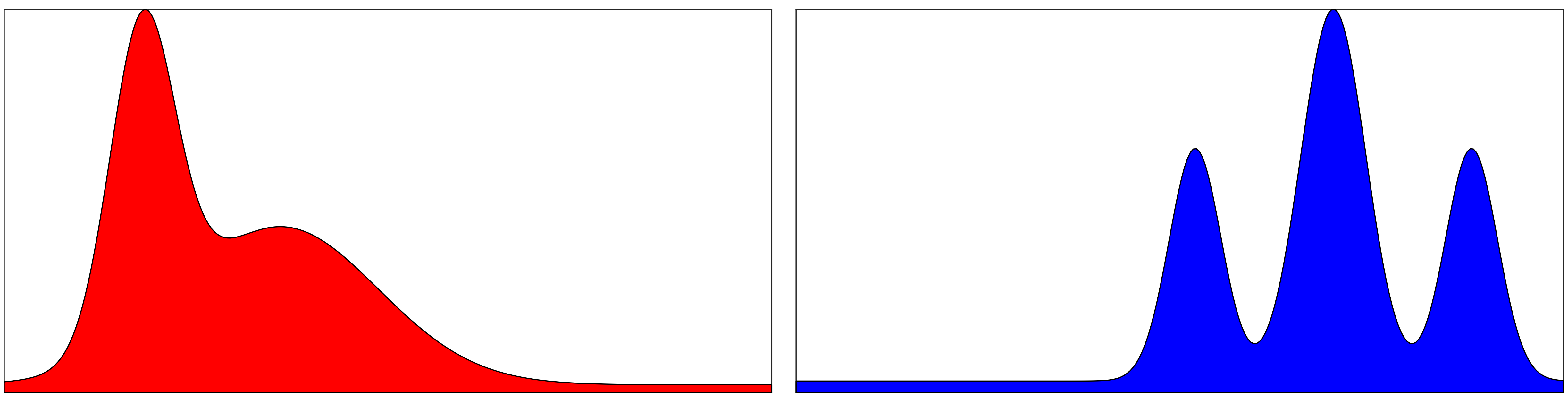
Remark. If $\Omega = \mathbb{R}$, $\boldsymbol{c}(x, y) = \boldsymbol{c}(|x - y|)$, $\boldsymbol{c}$ convex, $F_{\boldsymbol{\mu}}^{-1}, F_{\boldsymbol{\nu}}^{-1}$ quantile functions,

$$W(\boldsymbol{\mu}, \boldsymbol{\nu}) = \int_0^1 \boldsymbol{c}(|F_{\boldsymbol{\mu}}^{-1}(x) - F_{\boldsymbol{\nu}}^{-1}(x)|)dx$$

Recently: projection as regularization

Remark. If $\Omega = \mathbb{R}$, $\boldsymbol{c}(x, y) = \boldsymbol{c}(|x - y|)$, $\boldsymbol{c}$ convex, $F_{\boldsymbol{\mu}}^{-1}, F_{\boldsymbol{\nu}}^{-1}$ quantile functions,

$$W(\boldsymbol{\mu}, \boldsymbol{\nu}) = \int_0^1 \boldsymbol{c}(|F_{\boldsymbol{\mu}}^{-1}(x) - F_{\boldsymbol{\nu}}^{-1}(x)|)dx$$

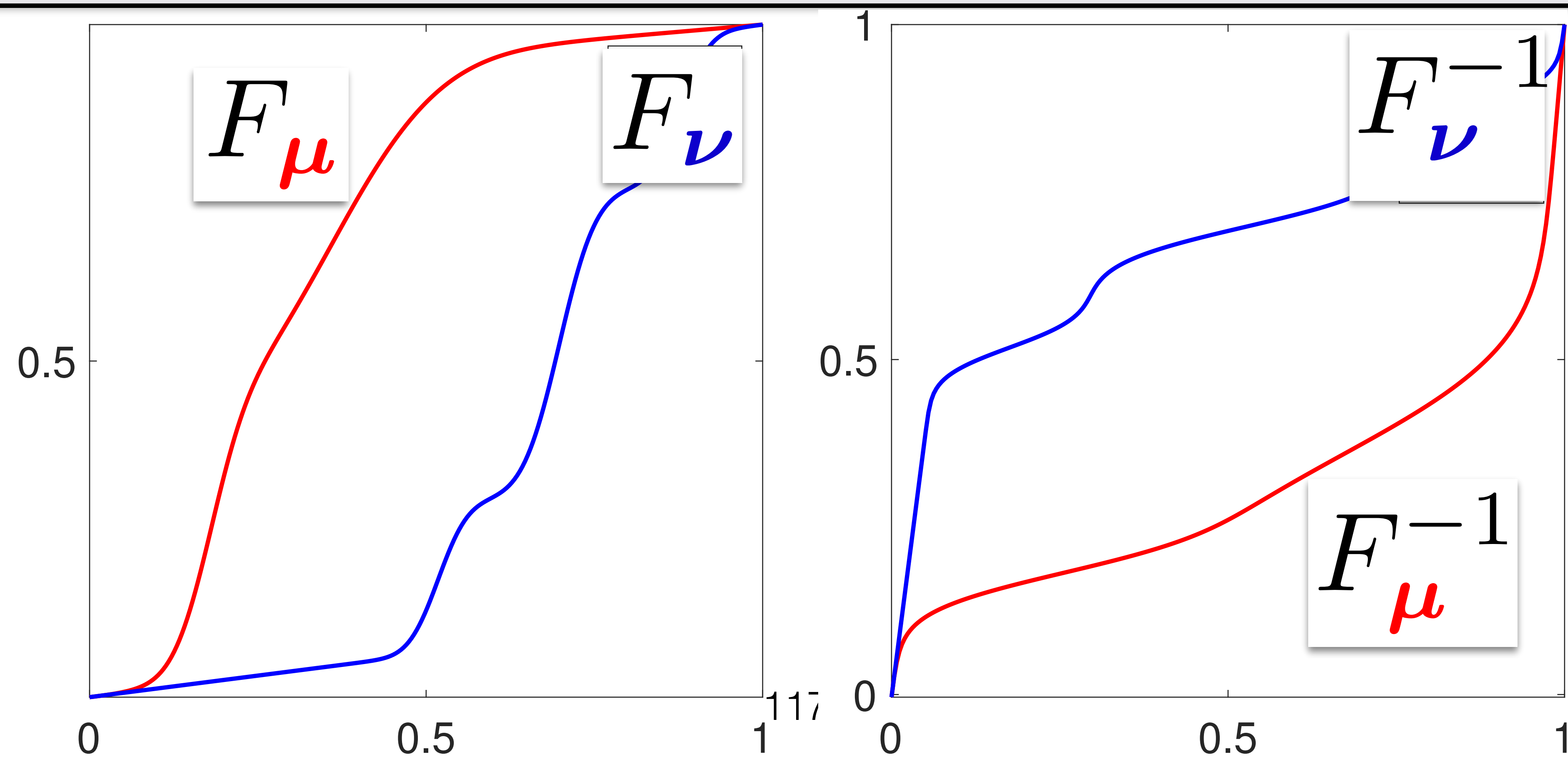


$\boldsymbol{\mu}$ $\boldsymbol{\nu}$

Recently: projection as regularization

Remark. If $\Omega = \mathbb{R}$, $\boldsymbol{c}(x,y) = \boldsymbol{c}(|x-y|)$, $\boldsymbol{c}$ convex, $F_{\boldsymbol{\mu}}^{-1}, F_{\boldsymbol{\nu}}^{-1}$ quantile functions,

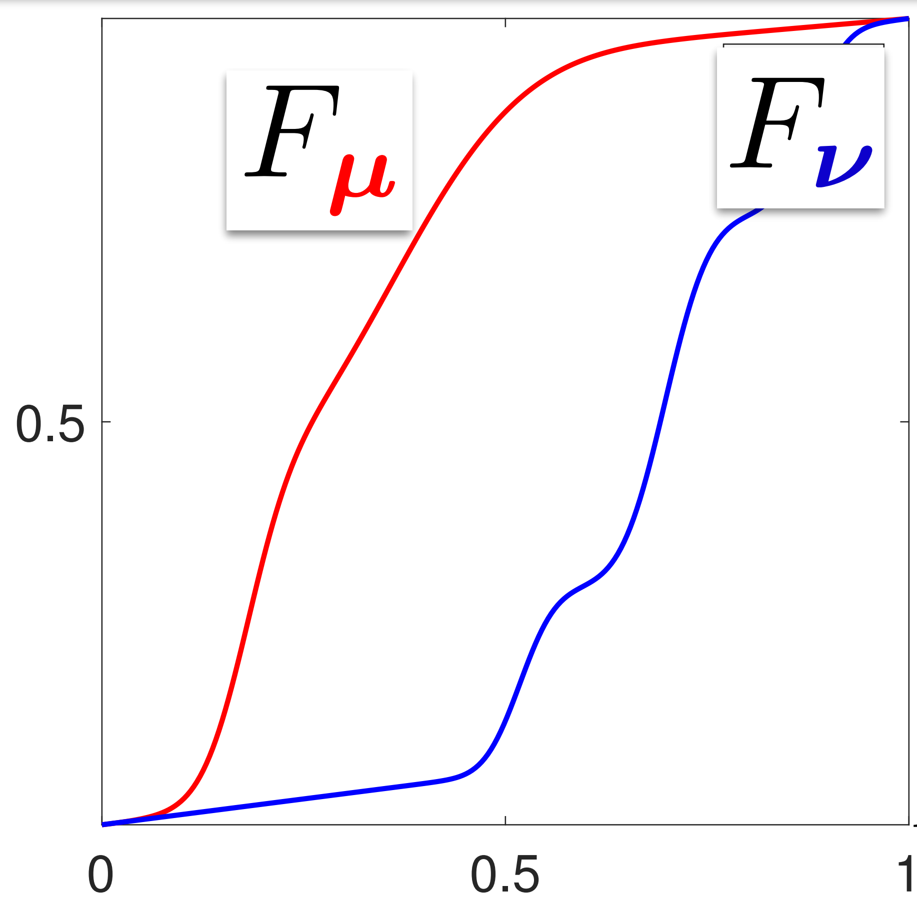
$$W(\boldsymbol{\mu},\boldsymbol{\nu}) = \int_0^1 \boldsymbol{c}(|F_{\boldsymbol{\mu}}^{-1}(x) - F_{\boldsymbol{\nu}}^{-1}(x)|)dx$$

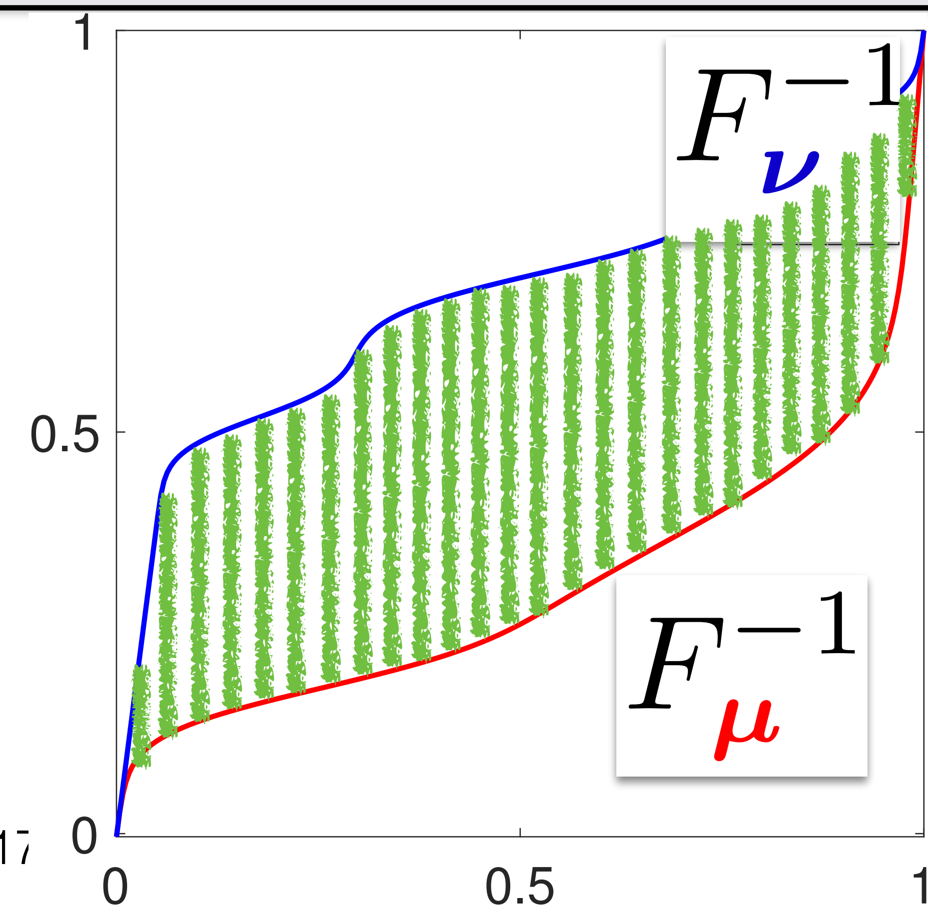


Recently: projection as regularization

Remark. If $\Omega = \mathbb{R}$, $\boldsymbol{c}(x,y) = \boldsymbol{c}(|x-y|)$, $\boldsymbol{c}$ convex, $F_{\boldsymbol{\mu}}^{-1}, F_{\boldsymbol{\nu}}^{-1}$ quantile functions,

$$W(\boldsymbol{\mu},\boldsymbol{\nu}) = \int_0^1 \boldsymbol{c}(|F_{\boldsymbol{\mu}}^{-1}(x) - F_{\boldsymbol{\nu}}^{-1}(x)|)dx$$





Recently: projection as regularization

Remark. If $\Omega = \mathbb{R}$, $\boldsymbol{c}(x, y) = \boldsymbol{c}(|x - y|)$, $\boldsymbol{c}$ convex, $F_{\boldsymbol{\mu}}^{-1}, F_{\boldsymbol{\nu}}^{-1}$ quantile functions,

$$W(\boldsymbol{\mu}, \boldsymbol{\nu}) = \int_0^1 \boldsymbol{c}(|F_{\boldsymbol{\mu}}^{-1}(x) - F_{\boldsymbol{\nu}}^{-1}(x)|)dx$$

Sliced Wasserstein Distance [Rabin+'11]

$$SW(\boldsymbol{\mu}, \boldsymbol{\nu}) = \mathbb{E}_{\theta \sim \mathcal{S}^{d-1}} \left[\int_0^1 \boldsymbol{c}(|F_{\theta^T_\sharp \boldsymbol{\mu}}^{-1}(x) - F_{\theta^T_\sharp \boldsymbol{\nu}}^{-1}(x)|)dx \right]$$

On k-dim (robust) projections

- Most often, PCA used to project data.

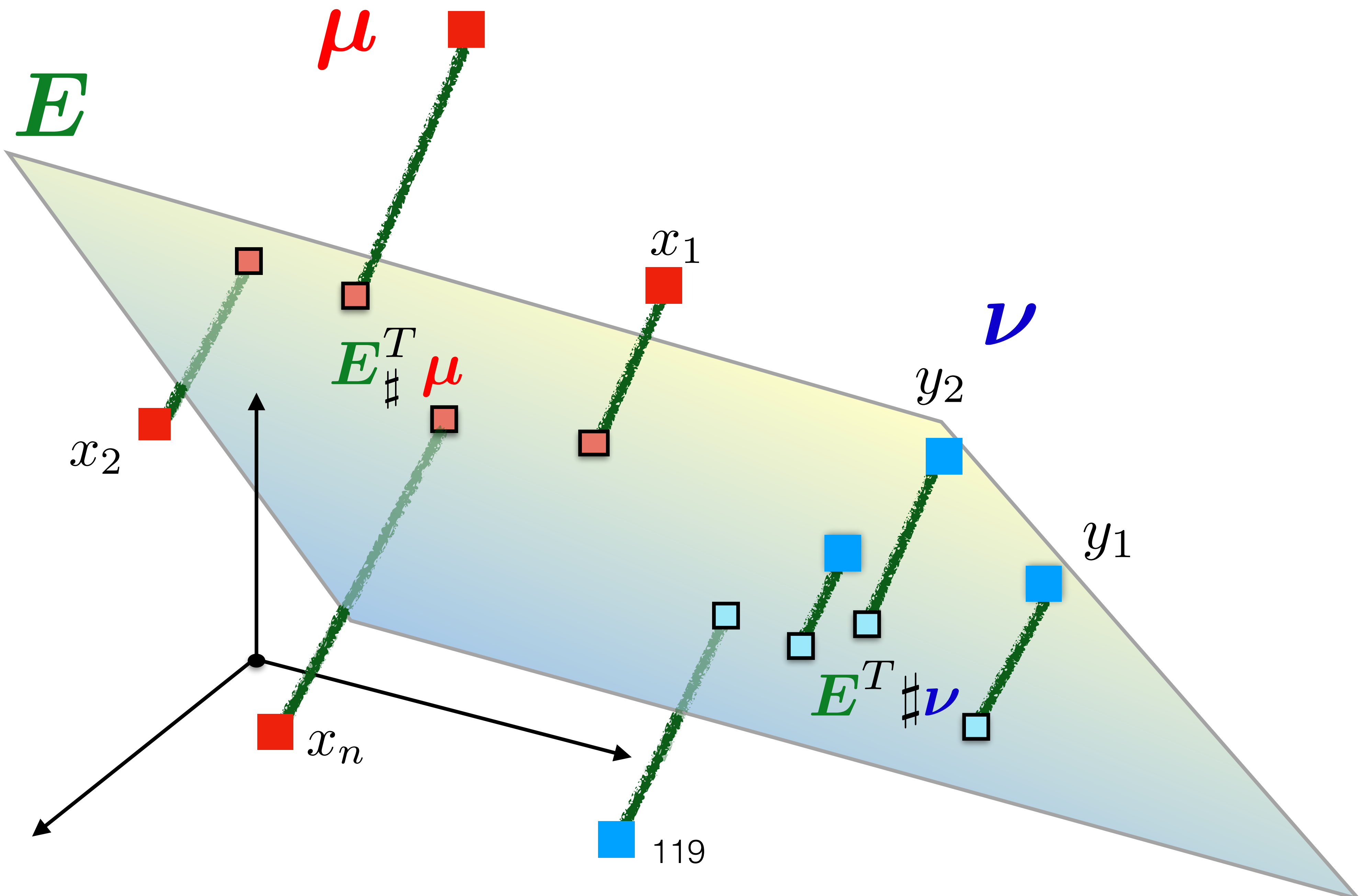
$$\tilde{W}_{p,k}(\boldsymbol{\mu}, \boldsymbol{\nu}) = W_p(\hat{\boldsymbol{L}}_{\boldsymbol{k}\sharp}\boldsymbol{\mu}, \hat{\boldsymbol{L}}_{\boldsymbol{k}\sharp}\boldsymbol{\nu})$$

- Look instead for "worst" possible projection.

$$\boxed{\mathcal{W}_{p,k}(\boldsymbol{\mu}, \boldsymbol{\nu}) = \max_{\substack{\boldsymbol{E}\in\mathbb{R}^{d\times k} \\ \boldsymbol{E}^T\boldsymbol{E}=I_k}} W_p(\boldsymbol{E}^T_\sharp \boldsymbol{\mu}, \boldsymbol{E}^T_\sharp \boldsymbol{\nu})}$$

On k-dim (robust) projections

$$\mathcal{W}_{p,k}(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\substack{\boldsymbol{E}\in\mathbb{R}^{d\times k}\\ \boldsymbol{E}^T\boldsymbol{E}=I_k}} W_p(\boldsymbol{E}^T_\sharp\boldsymbol{\mu},\boldsymbol{E}^T_\sharp\boldsymbol{\nu})$$



On k-dim (robust) projections

- Most often, PCA used to project data.

$$\tilde{W}_{p,k}(\boldsymbol{\mu},\boldsymbol{\nu}) = W_p(\hat{\boldsymbol{L}}_{\boldsymbol{k}\sharp}\boldsymbol{\mu}, \hat{\boldsymbol{L}}_{\boldsymbol{k}\sharp}\boldsymbol{\nu})$$

- Look instead for "worst" possible projection.

$$\boxed{\mathcal{W}_{p,k}(\boldsymbol{\mu},\boldsymbol{\nu}) = \max_{\substack{\boldsymbol{E}\in\mathbb{R}^{d\times k}\\ \boldsymbol{E}^T\boldsymbol{E}=I_k}} W_p(\boldsymbol{E}^T_\sharp\boldsymbol{\mu}, \boldsymbol{E}^T_\sharp\boldsymbol{\nu})}$$

- **[Weed+'19]:** convergence for "well behaved" measures effectively supported on k dimensional subspace, would recover estimator with $O(n^{-1/k})$

Not convex! min/max?

On k-dim (robust) projections

$$\mathcal{S}_{2,k} = \min_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \max_{\substack{\boldsymbol{E}\in\mathbb{R}^{d\times k}\\ \boldsymbol{E}^T\boldsymbol{E}=\boldsymbol{I_k}}} \iint \|\boldsymbol{E}^T x - \boldsymbol{E}^T y\|_2^2 \boldsymbol{P}(dx, dy)$$

On k-dim (robust) projections

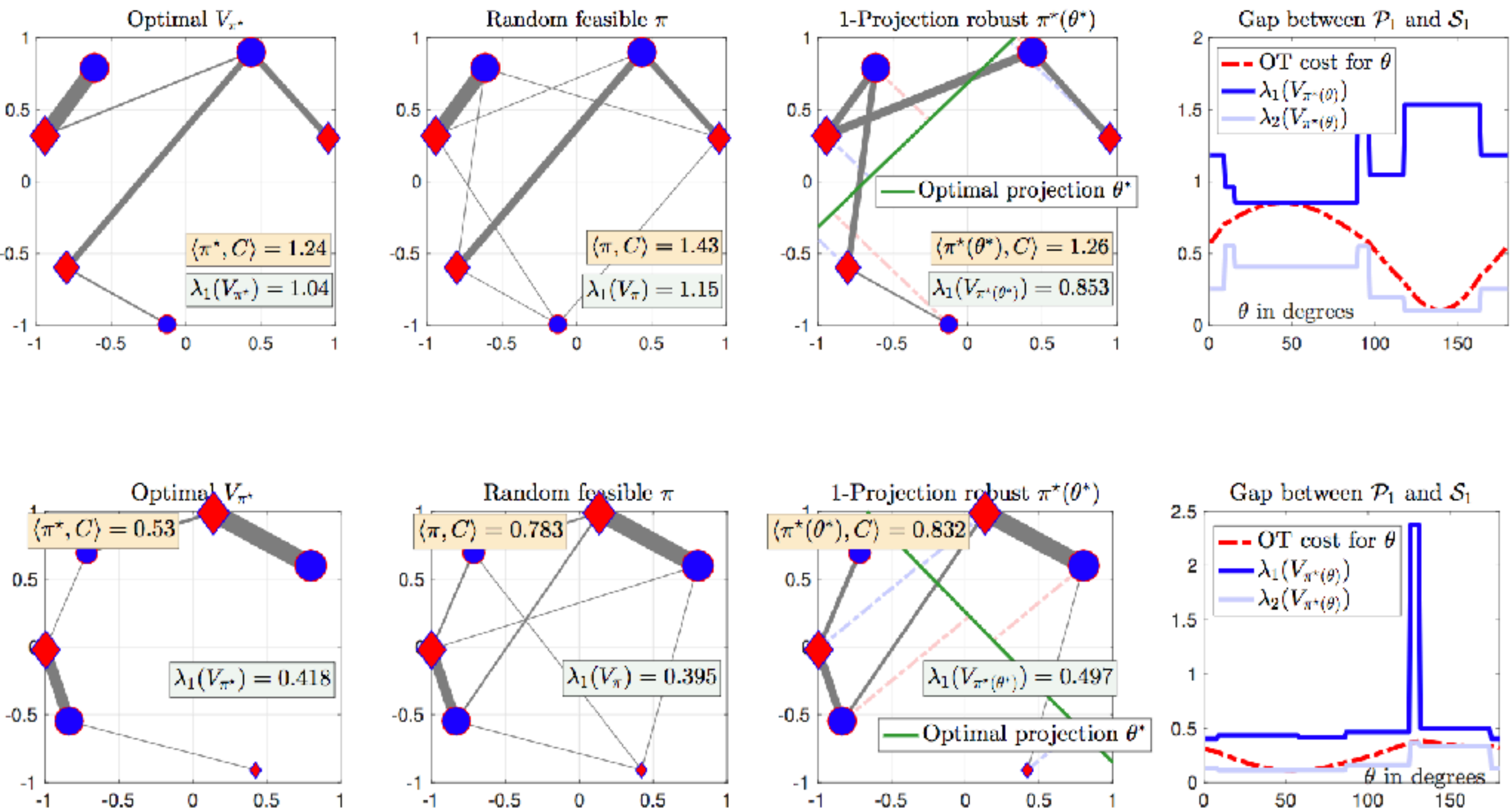
$$\mathcal{S}_{2,k} = \min_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \max_{\substack{\boldsymbol{E}\in\mathbb{R}^{d\times k}\\ \boldsymbol{E}^T\boldsymbol{E}=\boldsymbol{I_k}}} \iint \|\boldsymbol{E}^T x - \boldsymbol{E}^T y\|_2^2 \boldsymbol{P}(dx, dy)$$

$$\mathcal{S}_{2,k} = \min_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \max_{\substack{\boldsymbol{E}\in\mathbb{R}^{d\times k}\\ \boldsymbol{E}^T\boldsymbol{E}=\boldsymbol{I_k}}} \langle \boldsymbol{E}\boldsymbol{E}^T, V_{\boldsymbol{P}} \rangle$$

$$V_{\boldsymbol{P}} := \iint (x-y)(x-y)^T \boldsymbol{P}(dx, dy)$$

$$\mathcal{S}_{2,k} = \min_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \sum_{i=1}^{k} \lambda_i(V_{\boldsymbol{P}})$$

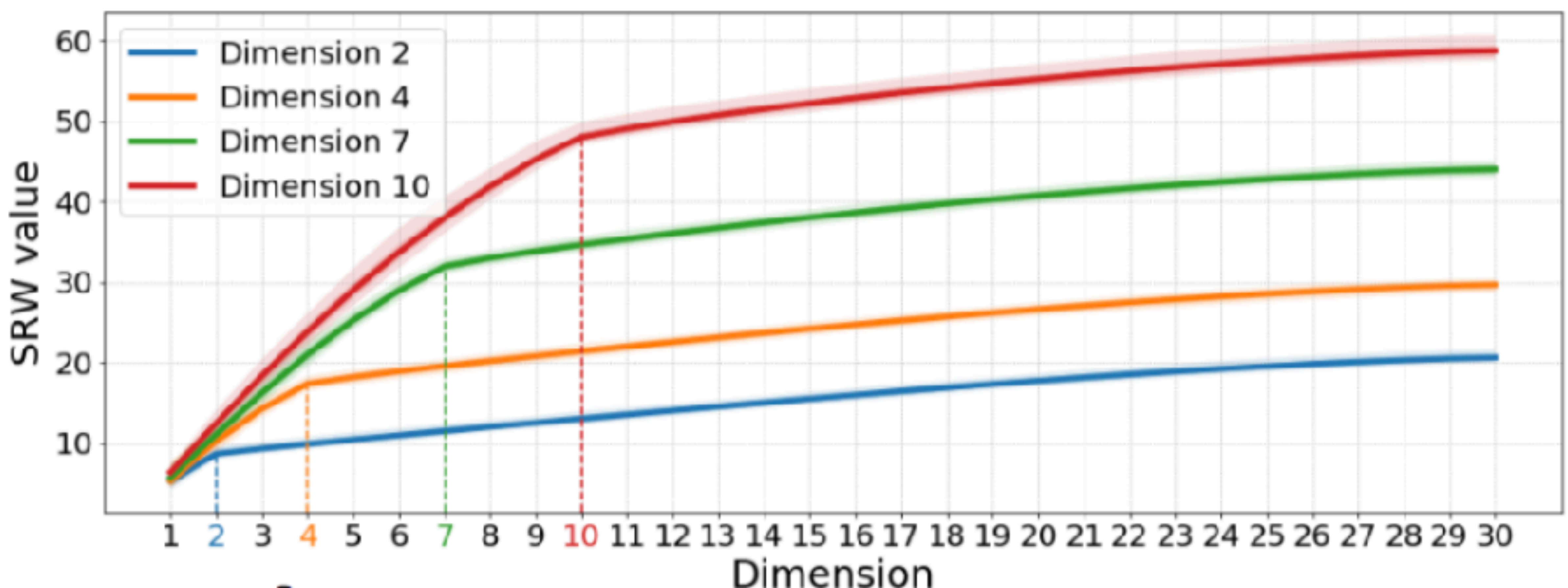
On k-dim robust projections



On k-dim robust projections

$$\mathcal{S}_{2,k} = \min_{\boldsymbol{P}\in\Pi(\boldsymbol{\mu},\boldsymbol{\nu})} \sum_{i=1}^{k} \lambda_i(V_{\boldsymbol{P}})$$

Solved using FW and entropic regularization (next)



4. Applications

- Wasserstein distances for retrieval
- Wasserstein barycenters
- W for unsupervised learning
- W inverse problems
- W in biology
- W to learn parameters and generative models

The Earth Mover's Distance

The Earth Mover's Distance

The Earth Mover's Distance

[Rubner'98] $\mathrm{dist}(I_1, I_2) = W_1(\boldsymbol{\mu}, \boldsymbol{\nu})$

The Word Mover's Distance

[Kusner'15] $\mathrm{dist}(D_1, D_2) = W_2(\boldsymbol{\mu}, \boldsymbol{\nu})$

Recall that our goal is...

Up to 2010: OT solvers used mostly for retrieval in databases of histograms

$$W_p(\boldsymbol{\mu}, \boldsymbol{\nu}) = ?$$

$$W_p(\boldsymbol{\mu}, \boldsymbol{\nu}) \leq \cdots ?$$

OT is now used as a **loss** or **fidelity** term:

$$\underset{\boldsymbol{\mu} \in \mathcal{P}(\Omega)}{\operatorname{argmin}} \; F(W_p(\boldsymbol{\mu}, \boldsymbol{\nu_1}), W_p(\boldsymbol{\mu}, \boldsymbol{\nu_2}), \dots, \boldsymbol{\mu}) = ?$$

[Jordan Kinderlehrer Otto'98]
[Ambrosio Gigli Savaré'05]

$$\text{“}\nabla_{\boldsymbol{\mu}}\text{”} W_p(\boldsymbol{\mu}, \boldsymbol{\nu}) = ?$$

Wassersteinization

[wos-ur-stahyn-ahy-sey-sh*uh*-n]
noun.

Introduction of optimal transport into an optimization or learning problem.

cf. least-squarification, L_1ification, deep-netification, kernelization

"Wasserstein + Data" Problems

- Quantization, k-means problem **[Lloyd'82]**

$$\min_{\substack{\boldsymbol{\mu}\in\mathcal{P}(\mathbb{R}^d)\\ |\operatorname{supp}\boldsymbol{\mu}|=k}} W_2^2(\boldsymbol{\mu}, \boldsymbol{\nu}_{\text{data}})$$

- **[McCann'95]** Interpolant

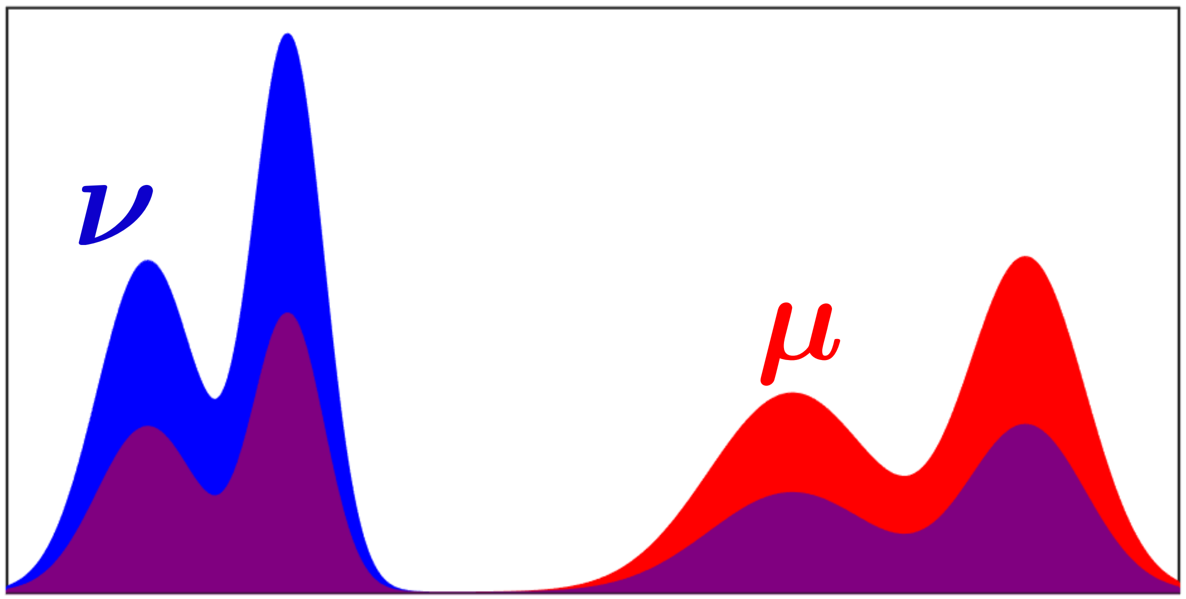
$$\min_{\boldsymbol{\mu}\in\mathcal{P}(\Omega)} (1-t)W_2^2(\boldsymbol{\mu}, \boldsymbol{\nu_1}) + tW_2^2(\boldsymbol{\mu}, \boldsymbol{\nu_2})$$

- **[JKO'98]** PDE's as gradient flows in $(\mathcal{P}(\Omega), W)$.

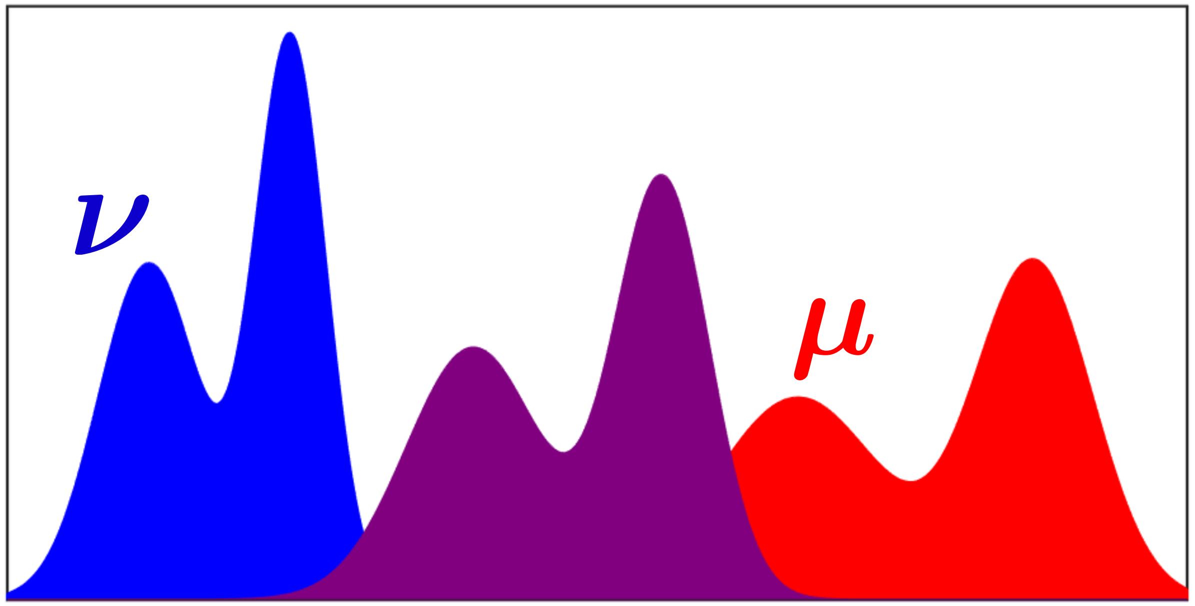
$$\mu_{t+1} = \operatorname*{argmin}_{\boldsymbol{\mu}\in\mathcal{P}(\Omega)} J(\boldsymbol{\mu}) + \lambda_t W_p^p(\boldsymbol{\mu}, \mu_t)$$

Averaging Measures

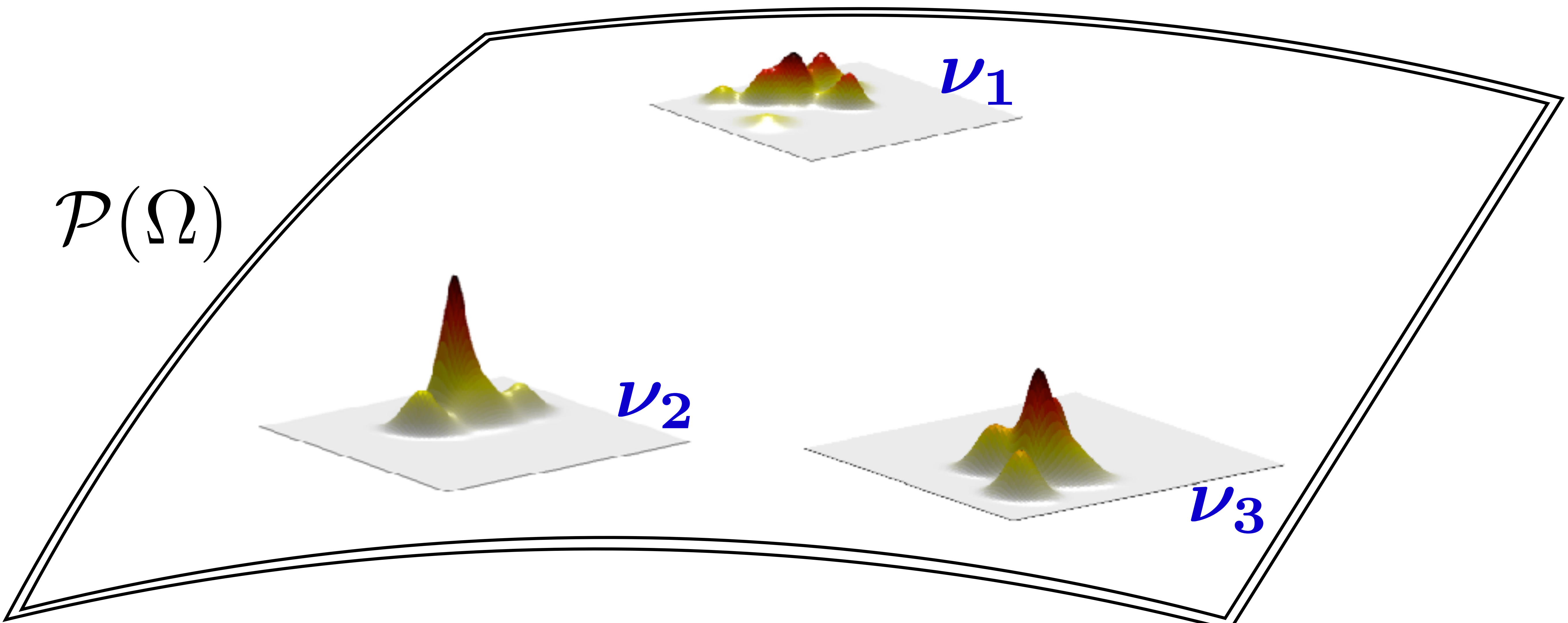
L_2 average



W average

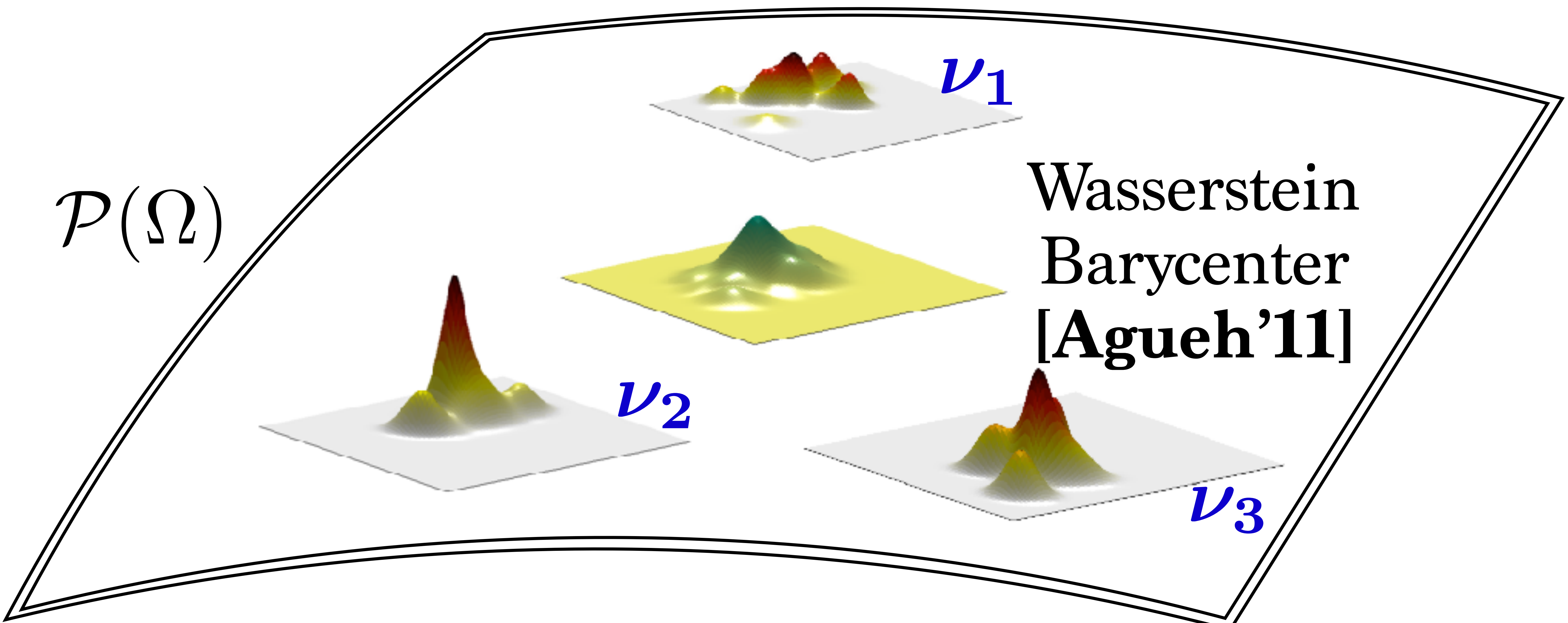


Barycenter for Measures?

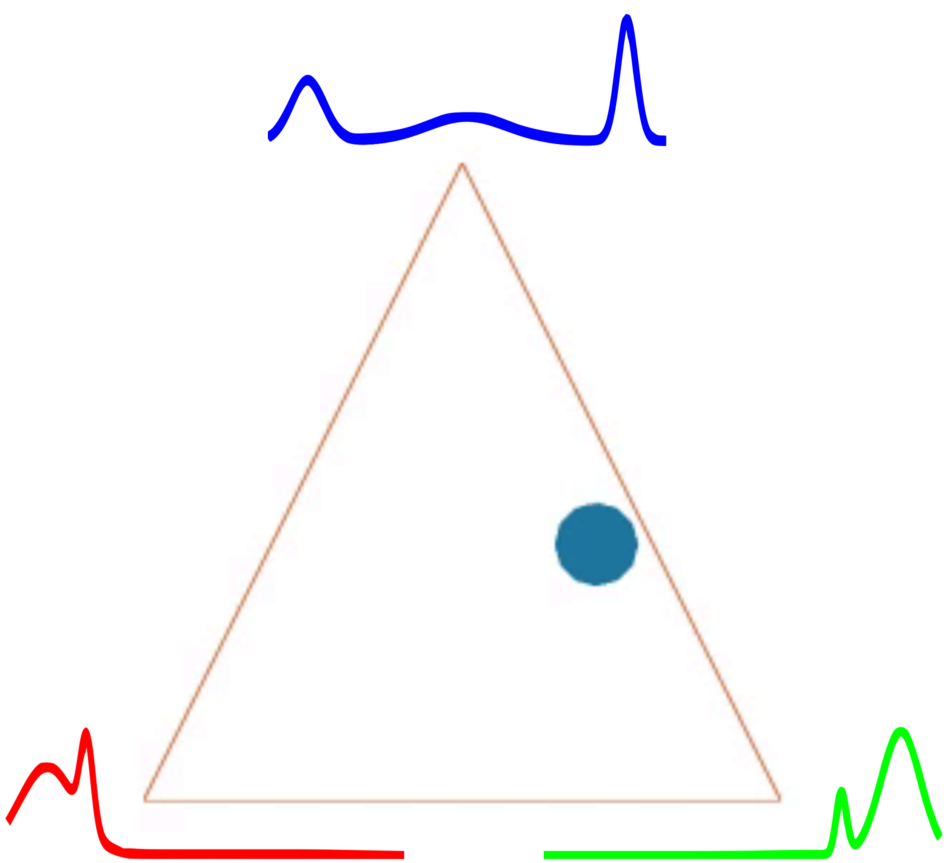


Barycenter for Measures?

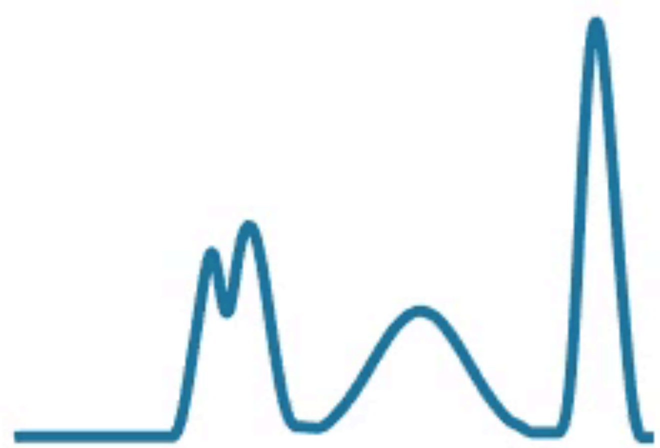
$$\min_{\mu \in \mathcal{P}(\Omega)} \sum_{i=1}^{N} \lambda_i W_p^p(\boldsymbol{\mu}, \boldsymbol{\nu_i})$$



Barycenter for Measures?



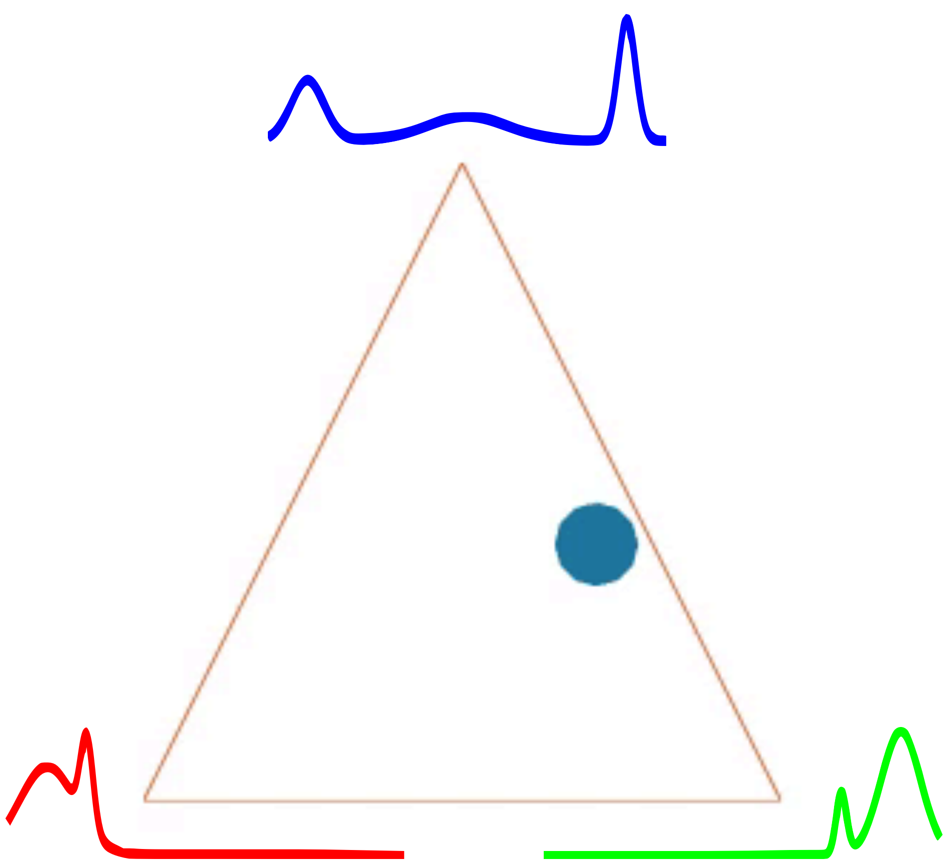
$\lambda \in \Sigma_3$



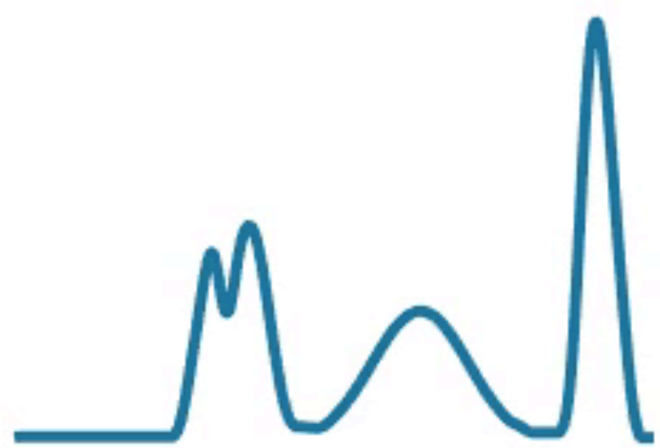
Wasserstein mean

L_2 mean

Barycenter for Measures?



$\lambda \in \Sigma_3$

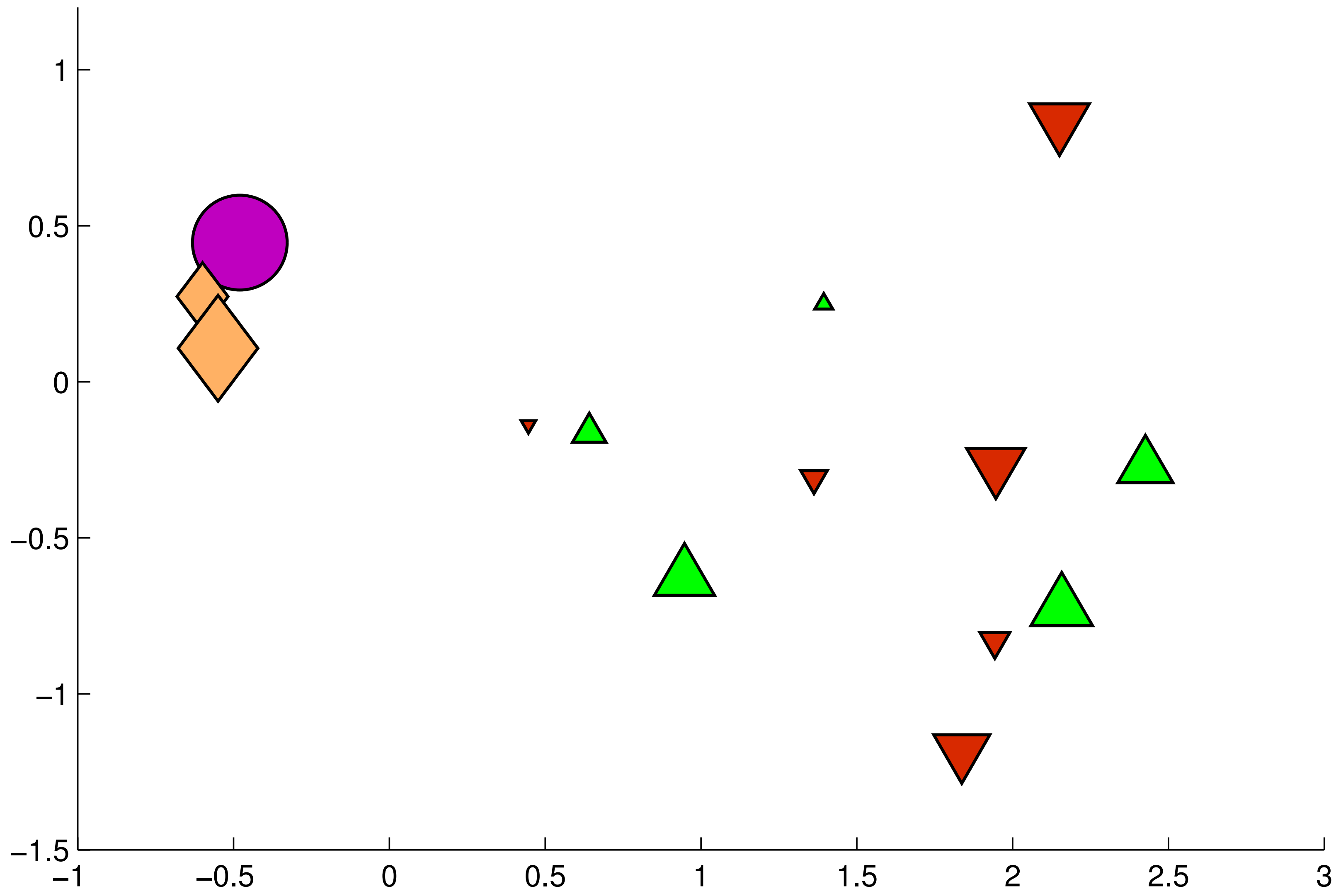


Wasserstein mean

L_2 mean

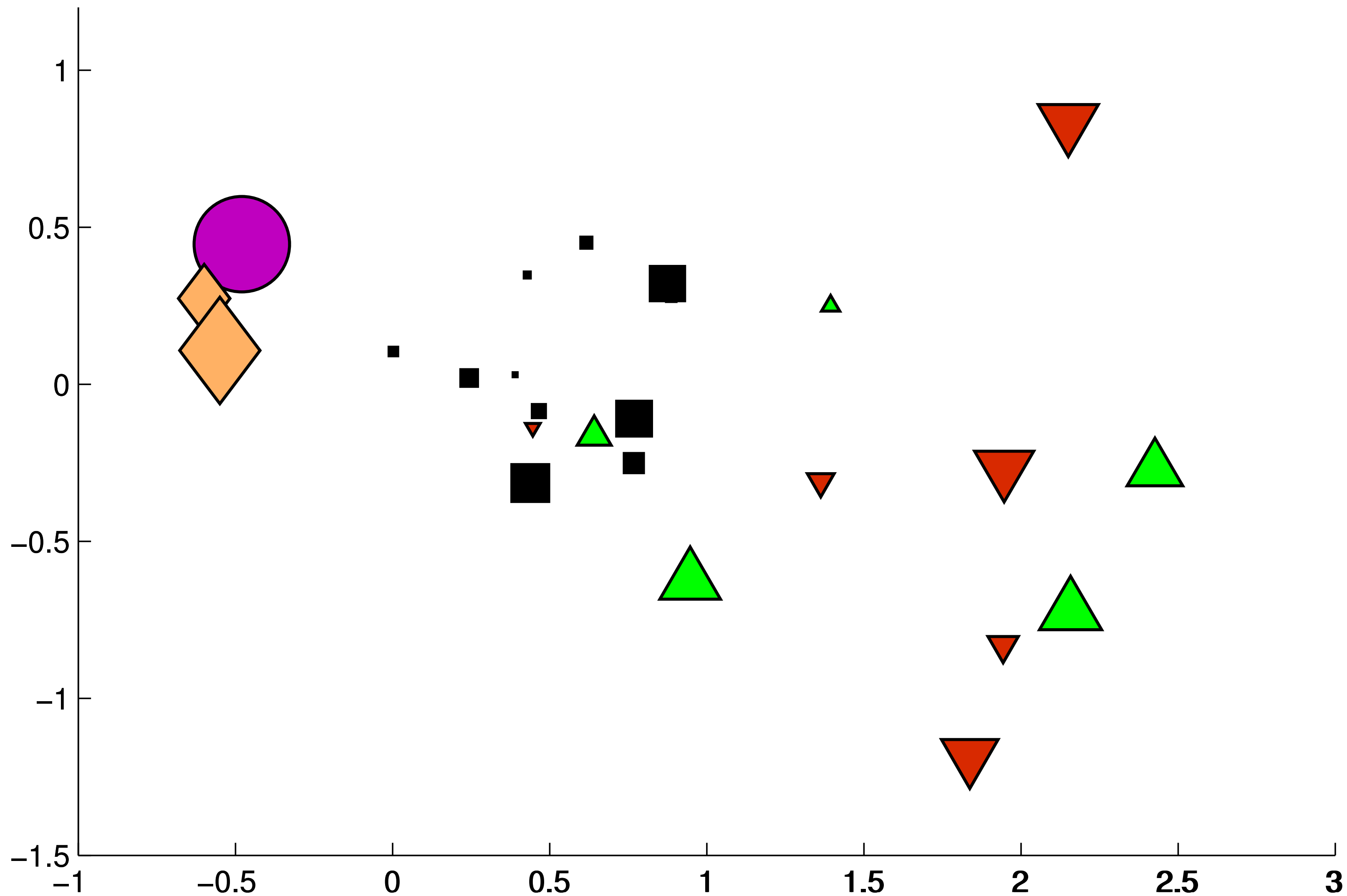
Multimarginal Formulation

- **Exact** solution (W_2) using MM-OT. **[Agueh'11]**



Multimarginal Formulation

- **Exact** solution (W_2) using MM-OT. **[Agueh'11]**



$$\text{If } |\operatorname{supp} \boldsymbol{\nu_i}| = \boldsymbol{n_i}, \text{ LP of size } \left(\prod_i \boldsymbol{n_i}, \sum_i \boldsymbol{n_i}\right)$$

Averaging Histograms is a LP

When Ω is a finite metric space defined by M.

$$\min_{\boldsymbol{a} \in \Sigma_n} \sum_i \lambda_i W_M(\boldsymbol{a}, \boldsymbol{b_i})$$

Averaging Histograms is a LP

When Ω is a finite metric space defined by M.

$$\min_{\boldsymbol{P_1},\cdots,\boldsymbol{P_N},\boldsymbol{a}} \sum_{i=1}^{N} \lambda_i \langle \boldsymbol{P_i}, M \rangle$$

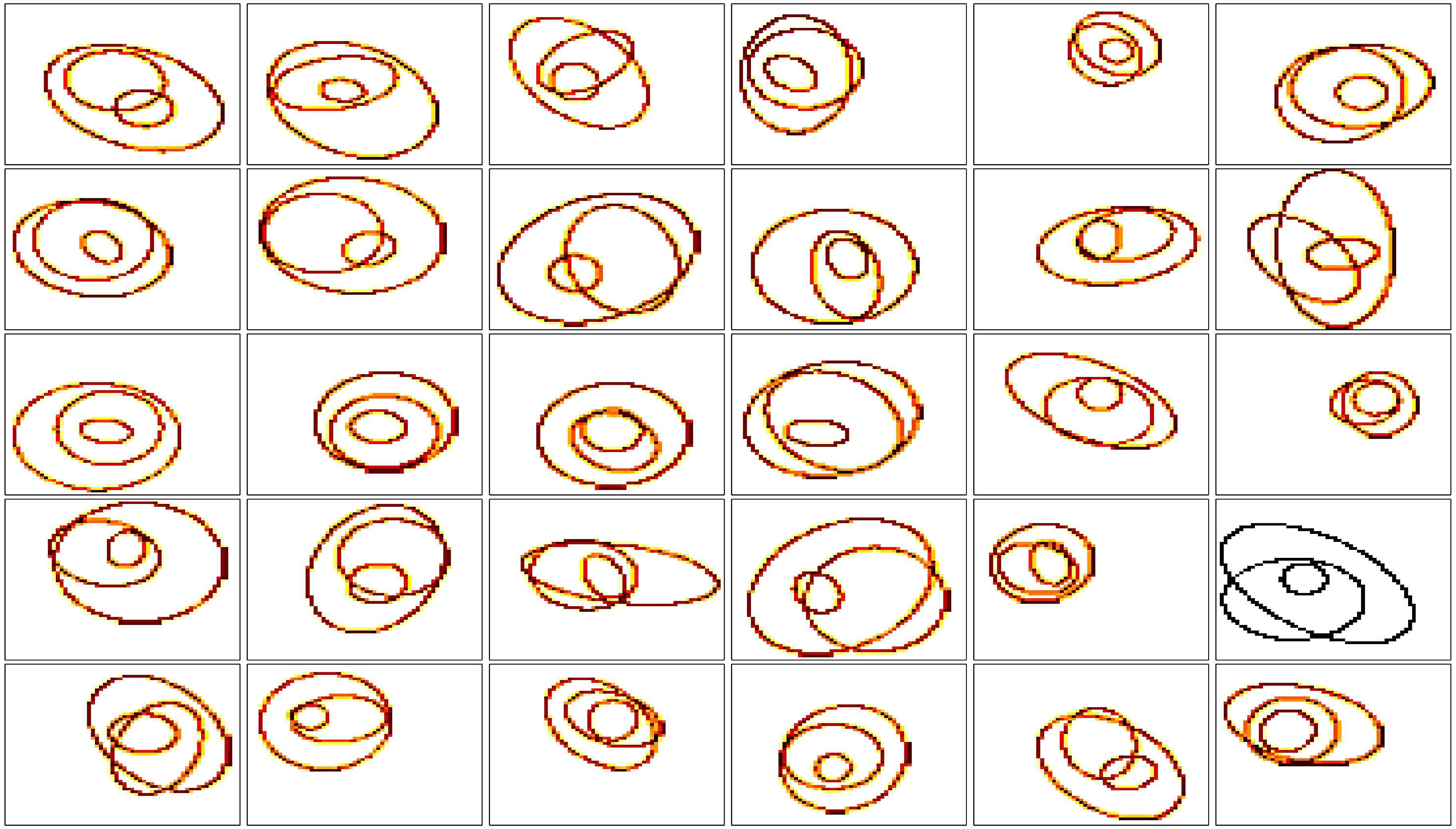
$$\text{s.t. } \boldsymbol{P_i}^T \mathbf{1}_n = \boldsymbol{b_i}, \forall i \leq N,$$

$$\boldsymbol{P_1} \mathbf{1}_n = \cdots = \boldsymbol{P_N} \mathbf{1}_d = \boldsymbol{a}.$$

If $|\Omega| = n$, LP of size $(Nn^2, (2N-1)n)$.

Primal Descent on Regularized W

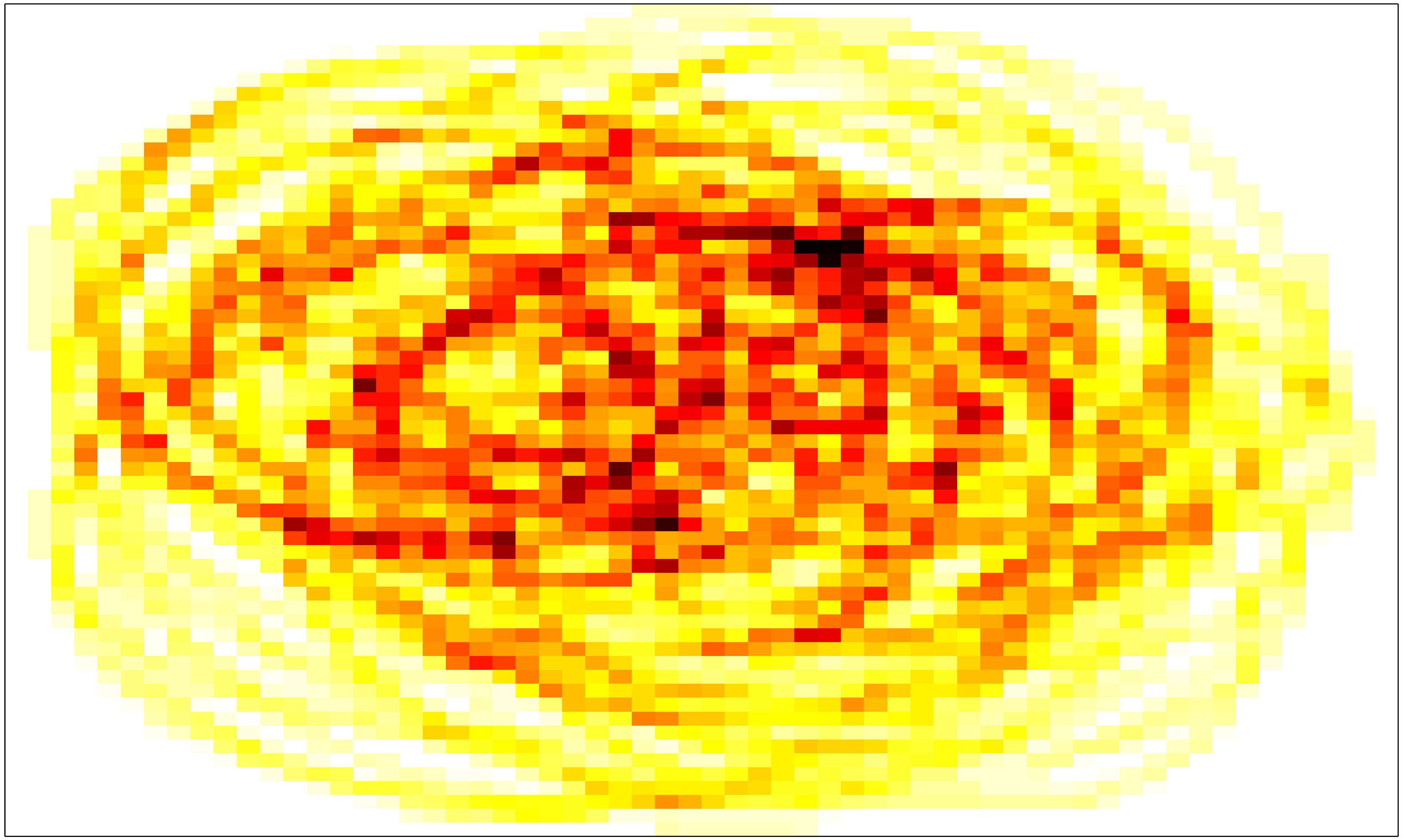
$$\min_{\boldsymbol{a} \in \Sigma_{h \times h}} \sum_{i=1}^{N} \lambda_i W_{\gamma}(\boldsymbol{a}, \boldsymbol{b_i})$$



[Cuturi'14]

Primal Descent on Regularized W

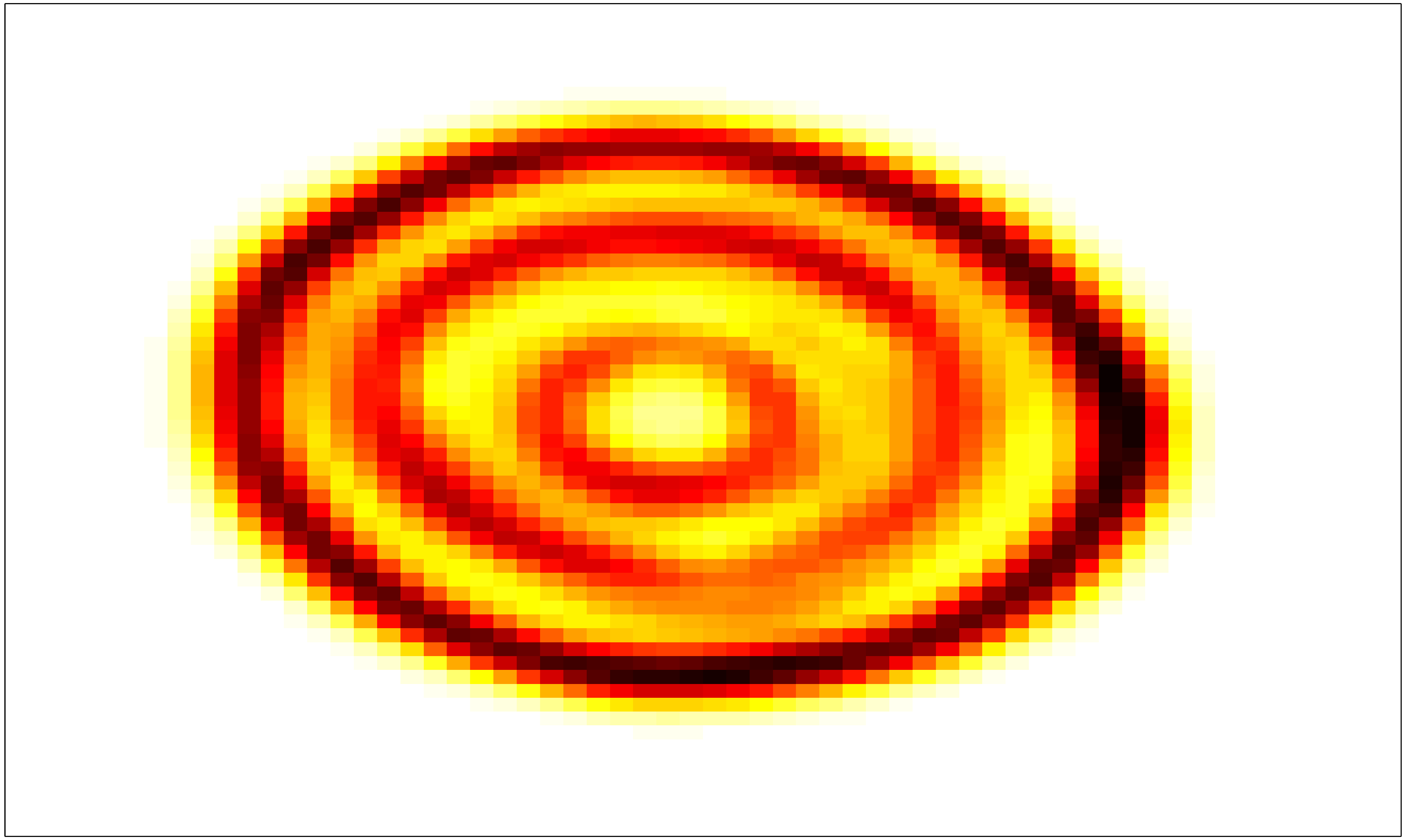
$$\min_{\mathbf{a}\in\Sigma_{h\times h}} \sum_{i=1}^{N} \lambda_i W_{\gamma}(\mathbf{a}, \mathbf{b_i})$$



[Cuturi'14]

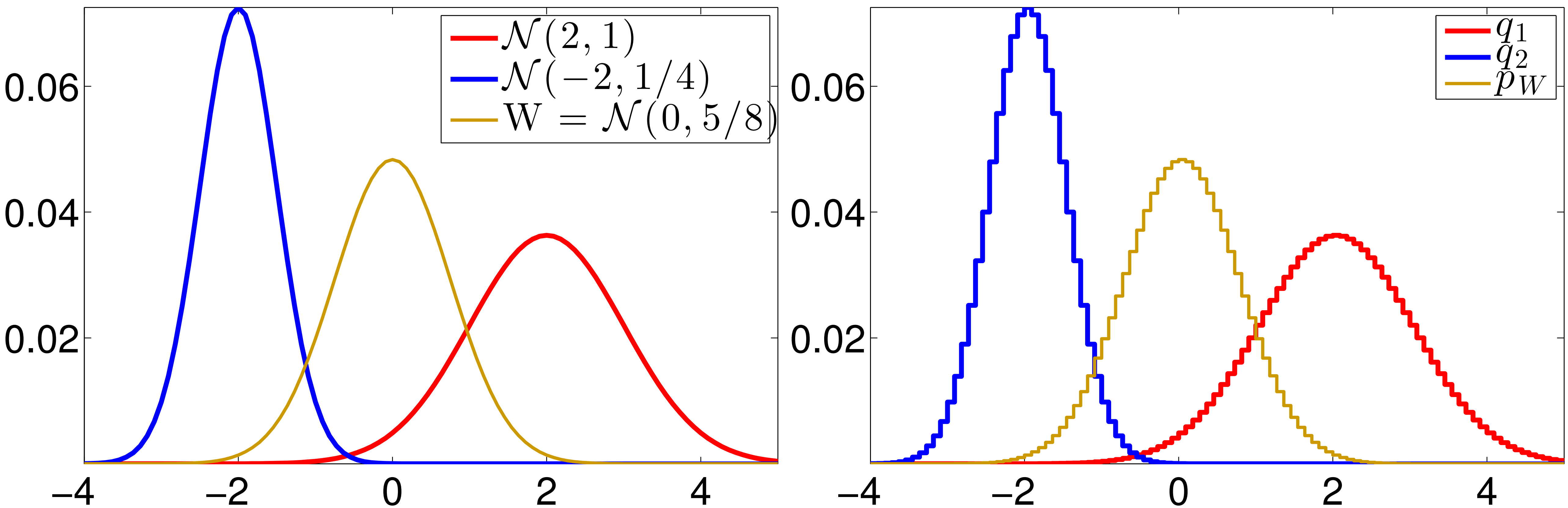
Primal Descent on Regularized W

$$\min_{\boldsymbol{a} \in \Sigma_{h \times h}} \sum_{i=1}^{N} \lambda_i W_{\gamma}(\boldsymbol{a}, \boldsymbol{b_i})$$

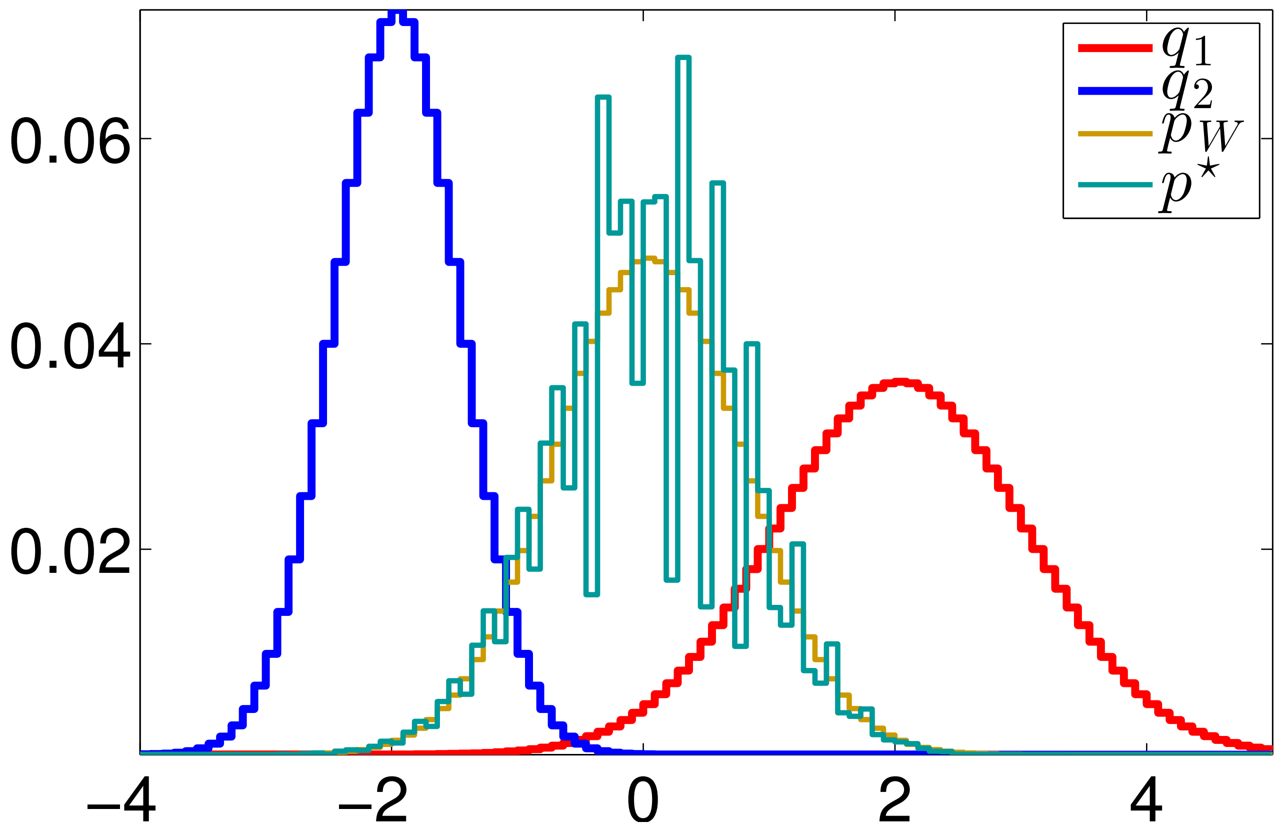


[Cuturi'14]

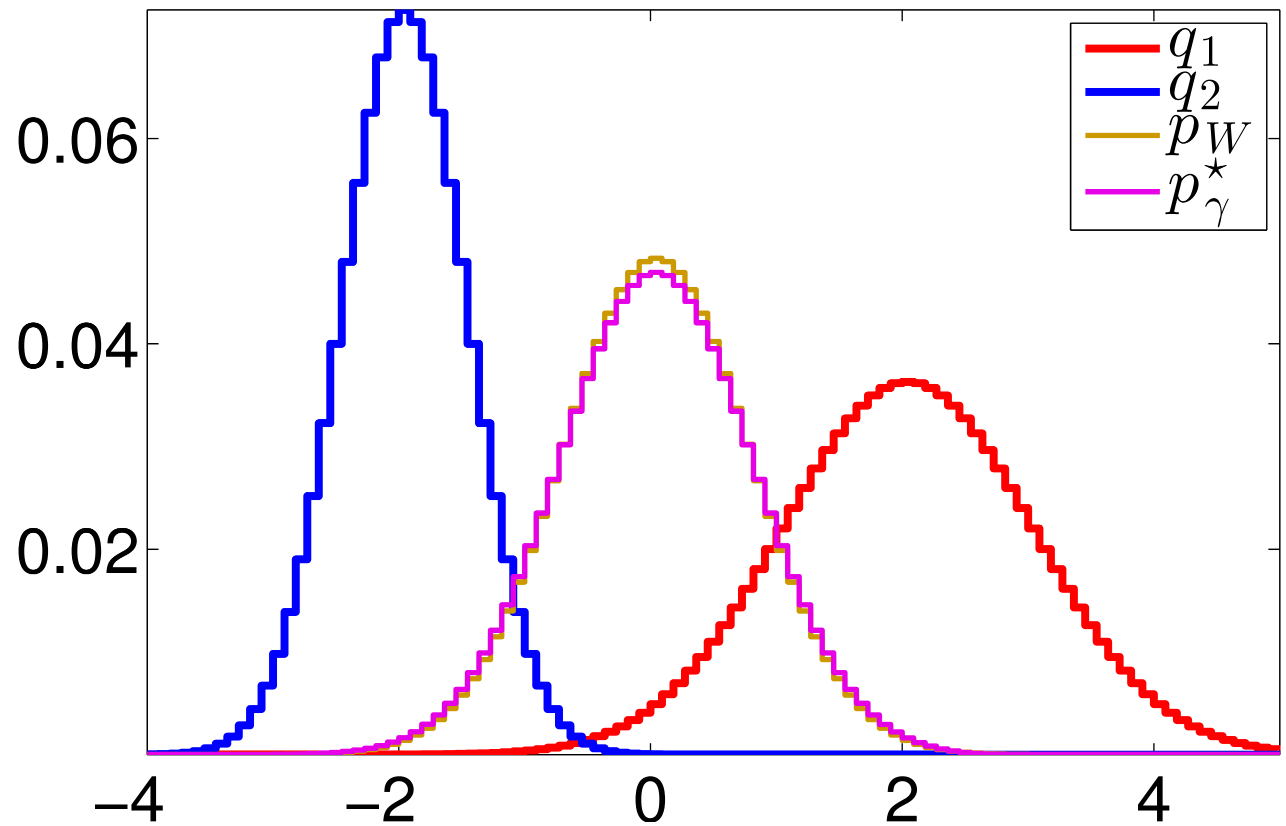
On Regularizing or Not



On Regularizing or Not

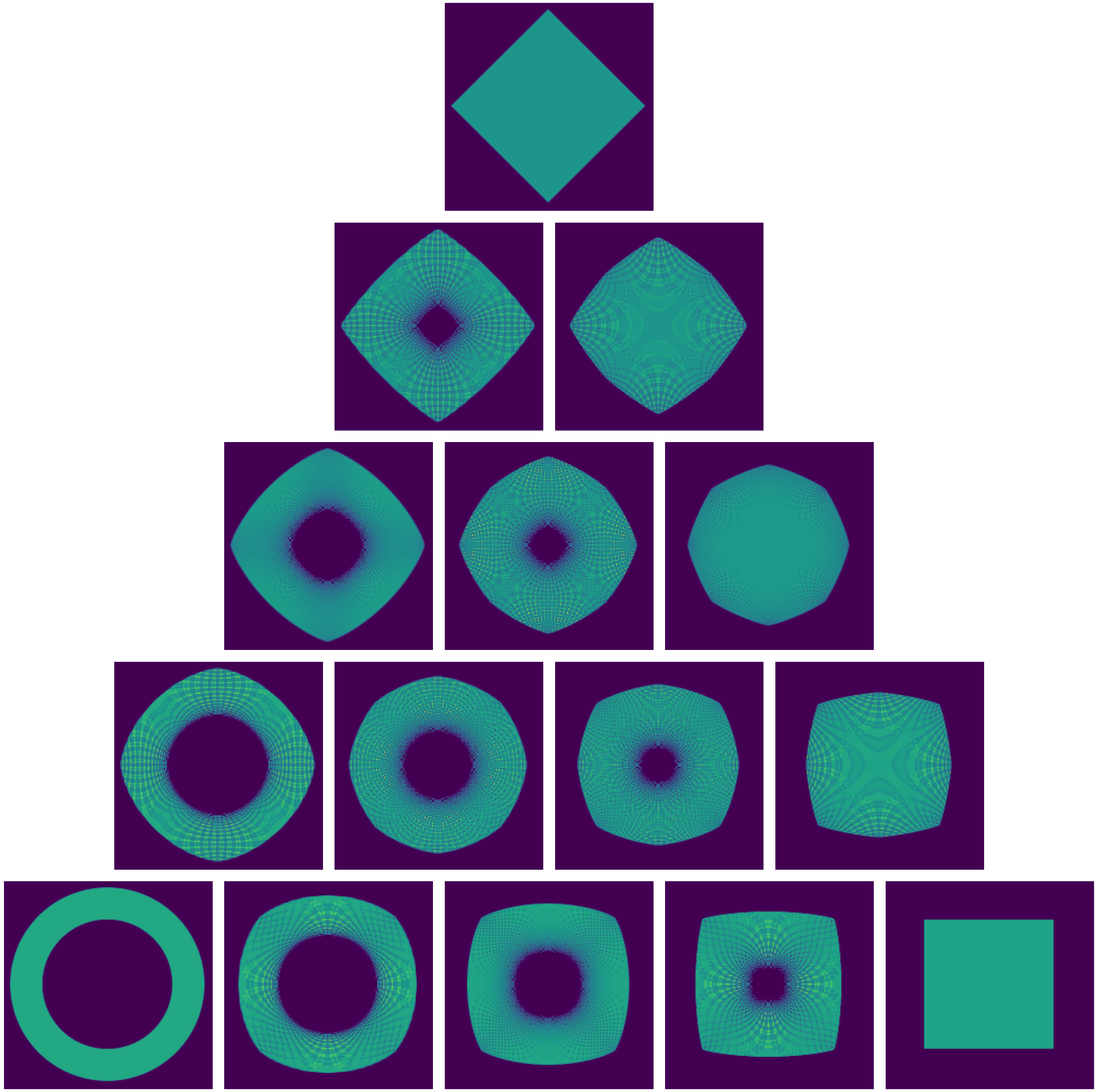


True barycenter



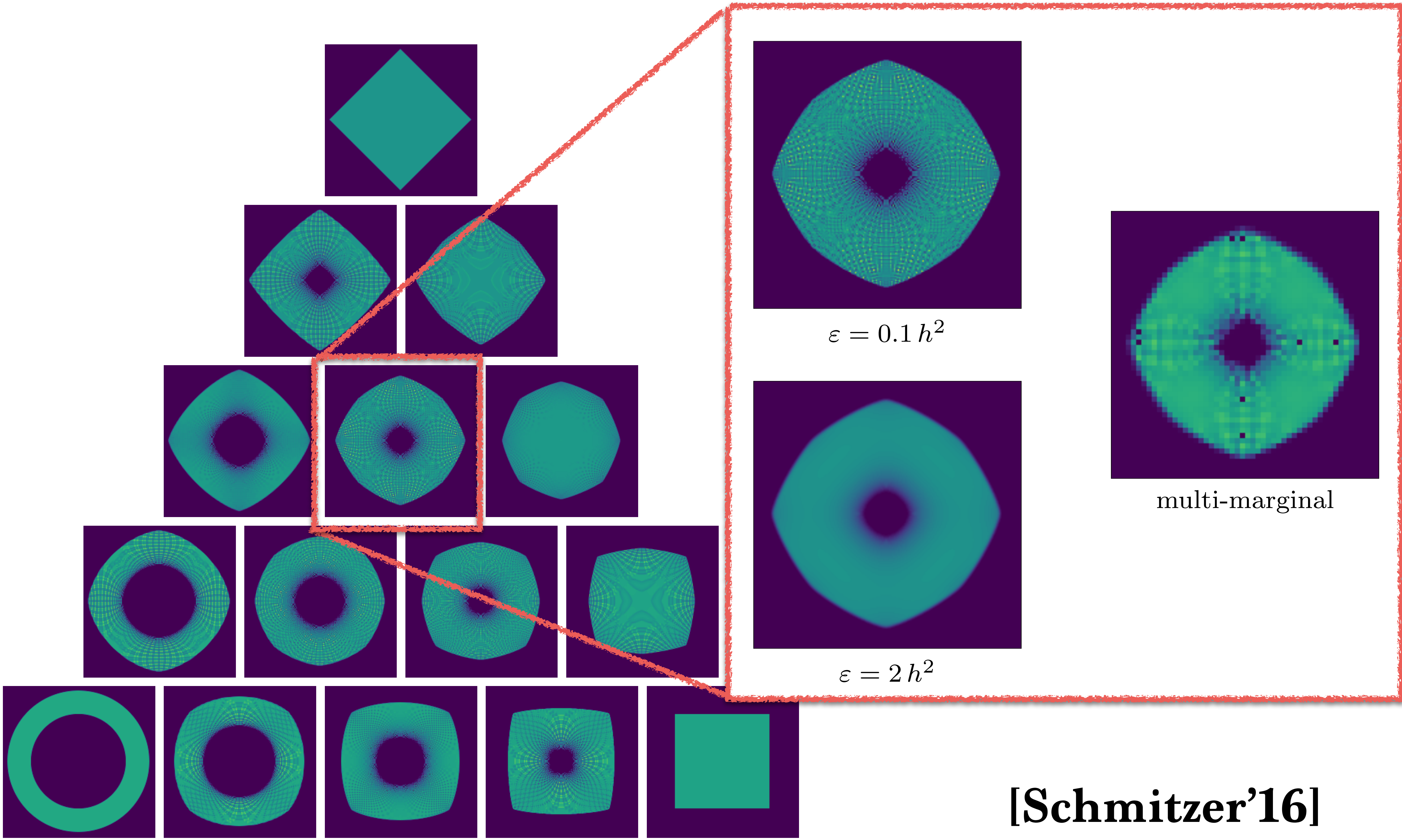
Barycenter using regularized OT

On Regularizing or Not



[Schmitzer'16]

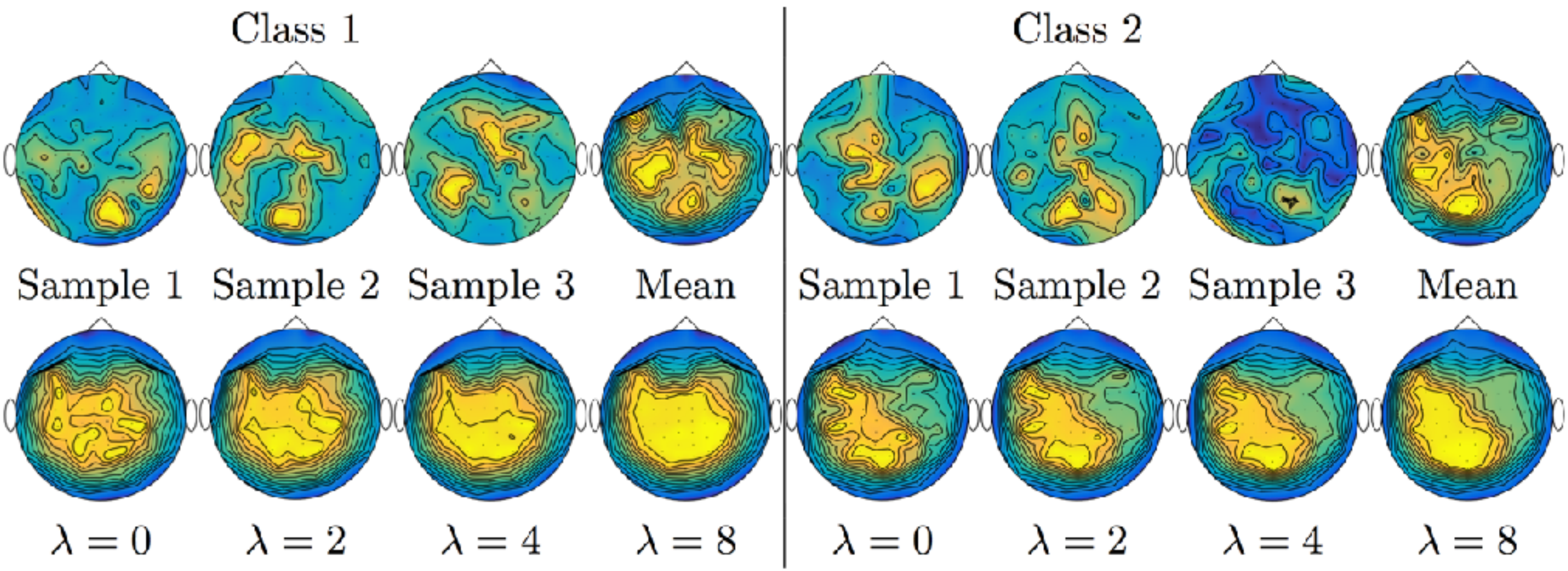
On Regularizing or Not



[Schmitzer'16]

Duality: Regularized Barycenters

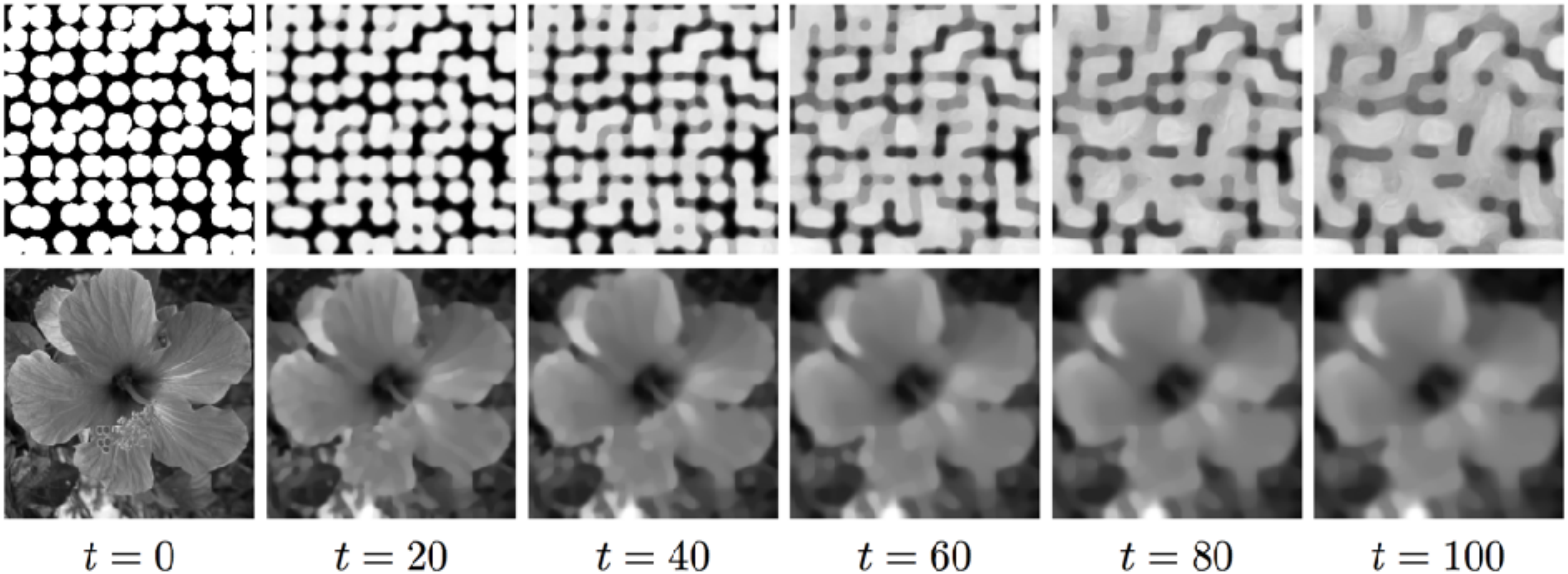
$$\min_{\mu \in \mathcal{P}(\Omega)} \sum_{i=1}^{N} W_{\gamma}(\mu, \nu_i) + \lambda \mathrm{TV}(\mu)$$



A Smoothed Dual Approach for Variational Wasserstein Problems
SIAM Imaging Sciences, 2016 **[CP'16]**

Duality: TV Gradient Flow

$$\mu_{k+1} = \underset{\mu \in \mathcal{P}(\Omega)}{\operatorname{argmin}} W_\gamma(\mu, \mu_k) + \tau \mathrm{TV}(\mu)$$



$t = k\tau$

A Smoothed Dual Approach for Variational Wasserstein Problems **[CP'16]**
SIAM Imaging Sciences, 2016

Regularized OT as KL Projection

$$\mathbf{KL}(P|\,K) = \sum_{ij} P_{ij} \log\left(P_{ij}/\,K_{ij}\right)$$

$$\langle P, M_{\mathbf{XY}}\rangle - \gamma E(P) = \gamma \mathbf{KL}(P|\,K)$$

$$\textbf{Prop.}\;\; P_\gamma = \mathrm{Proj}_{C_{\boldsymbol{a}} \cap C'_{\boldsymbol{b}}}(K)$$

$$C_{\boldsymbol{a}} = \{P | P\mathbf{1}_m = \boldsymbol{a}\}\,,\; C'_{\boldsymbol{b}} = \{P | P^T\mathbf{1}_n = \boldsymbol{b}\}$$

Regularized OT as KL Projection

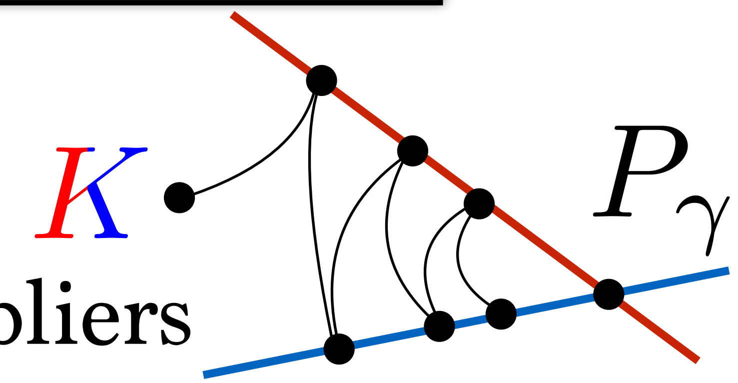
$$\textbf{Prop.}\ \ P_\gamma = \mathrm{Proj}_{C_{\boldsymbol{a}} \cap C'_{\boldsymbol{b}}}(K)$$

$$C_{\boldsymbol{a}} = \{P | P\mathbf{1}_m = \boldsymbol{a}\},\ C'_{\boldsymbol{b}} = \{P | P^T\mathbf{1}_n = \boldsymbol{b}\}$$

$$\mathrm{Proj}_{C_{\boldsymbol{a}}}(P) = \mathbf{D}\left(\frac{\boldsymbol{a}}{P\mathbf{1}_m}\right)P,$$

$$\mathrm{Proj}_{C'_{\boldsymbol{b}}}(P) = P\,\mathbf{D}\left(\frac{\boldsymbol{b}}{P^T\mathbf{1}_n}\right).$$

1. Sinkhorn = Dykstra's alternate projection
2. Only need to store & update diagonal multipliers



Wasserstein Barycenter = KL Projections

$$\langle P, M_{\mathbf{XY}} \rangle - \gamma E(P) = \gamma \mathbf{KL}(P | K)$$

$$\min_{\boldsymbol{a}} \sum_{i=1}^{N} \lambda_i W_\gamma(\boldsymbol{a}, \boldsymbol{b_i}) = \min_{\substack{\mathbf{P}=[\boldsymbol{P_1},\dots,\boldsymbol{P_N}] \\ \mathbf{P}\in \boldsymbol{C_1} \cap \boldsymbol{C_2}}} \sum_{i=1}^{N} \lambda_i \mathbf{KL}(\boldsymbol{P_i} | K)$$

$$\boldsymbol{C_1} = \{\mathbf{P} | \exists \boldsymbol{a}, \forall i, P_i \mathbf{1}_m = \boldsymbol{a}\}$$

$$\boldsymbol{C_2} = \{\mathbf{P} | \forall i, P_i^T \mathbf{1}_n = \boldsymbol{b_i}\}$$

Wasserstein Barycenter = KL Projections

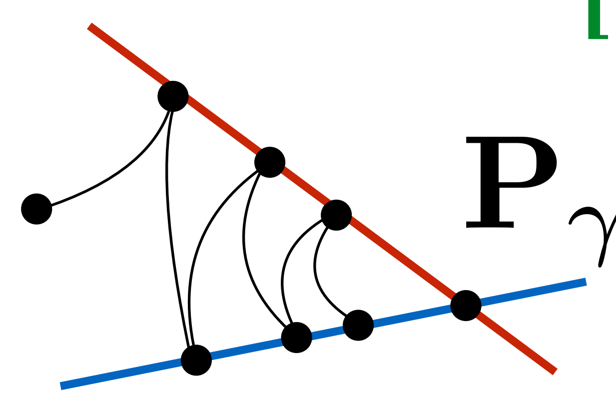
$$\min_{\boldsymbol{a}} \sum_{i=1}^{N} \lambda_i W_\gamma(\boldsymbol{a}, \boldsymbol{b_i}) = \min_{\substack{\mathbf{P}=[\boldsymbol{P_1},\ldots,\boldsymbol{P_N}] \\ \mathbf{P}\in \boldsymbol{C_1}\cap \boldsymbol{C_2}}} \sum_{i=1}^{N} \lambda_i \mathbf{KL}(\boldsymbol{P_i}|K)$$

$$\boldsymbol{C_1} = \{\mathbf{P} | \exists \boldsymbol{a}, \forall i, P_i \mathbf{1}_m = \boldsymbol{a}\}$$

$$\boldsymbol{C_2} = \left\{\mathbf{P} | \forall i, P_i^T \mathbf{1}_n = \boldsymbol{b_i}\right\}$$

[BCCNP'15]

$[\, K \cdots K\,]$



Wasserstein Barycenter = KL Projections

$$\min_{\boldsymbol{a}} \sum_{i=1}^{N} \lambda_i W_\gamma(\boldsymbol{a}, \boldsymbol{b_i}) = \min_{\substack{\mathbf{P}=[P_1,\ldots,P_N] \\ \mathbf{P}\in \boldsymbol{C_1} \cap \boldsymbol{C_2}}} \sum_{i=1}^{N} \lambda_i \mathbf{KL}(\boldsymbol{P_i} | K)$$

$$\boldsymbol{C_1} = \{\mathbf{P} | \exists \boldsymbol{a}, \forall i, P_i \mathbf{1}_m = \boldsymbol{a}\}$$

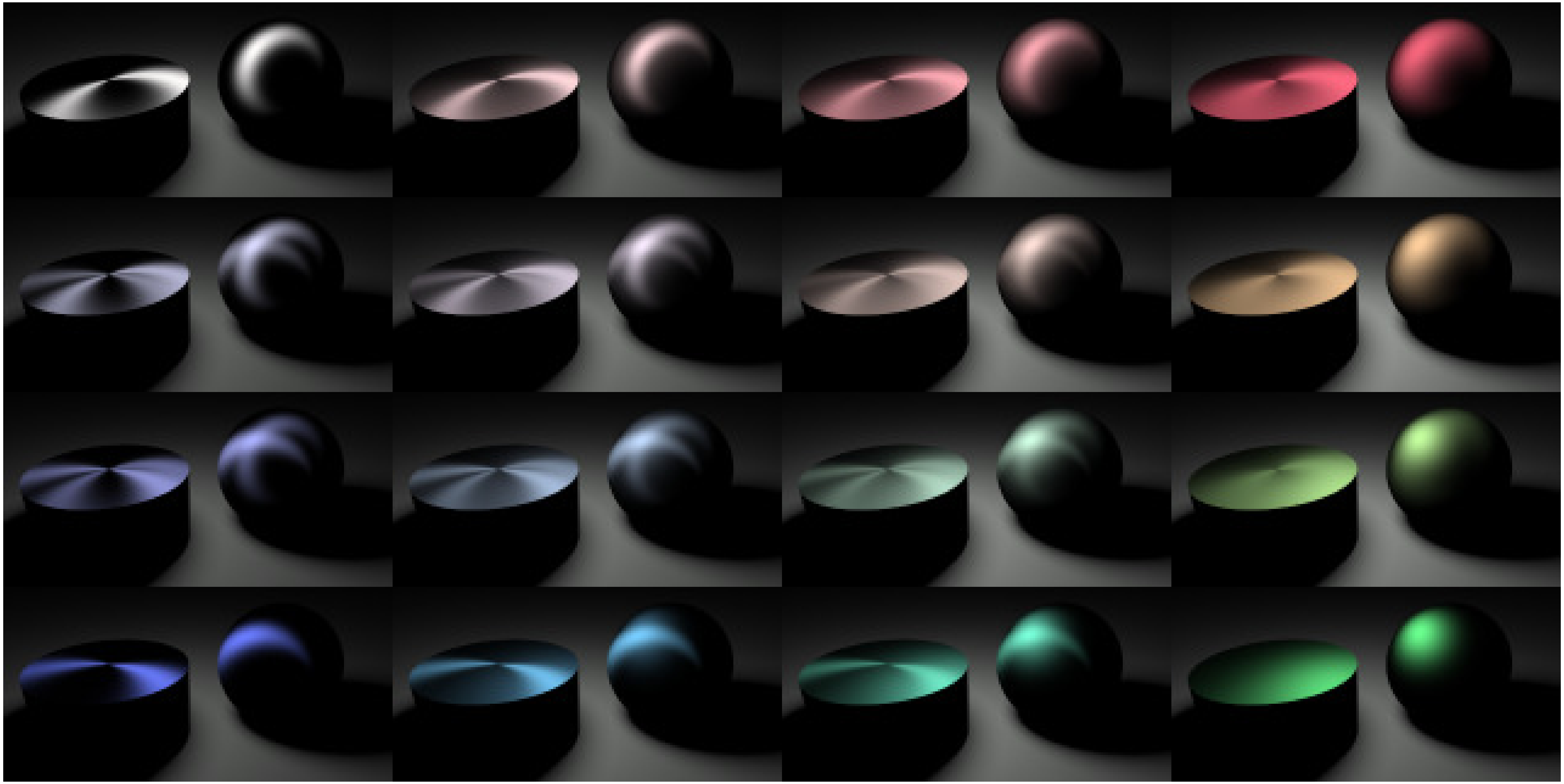
$$\boldsymbol{C_2} = \{\mathbf{P} | \forall i, P_i^T \mathbf{1}_n = \boldsymbol{b_i}\}$$

```
u=ones(size(B)); % d x N matrix
while not converged
   v=u.*(K'*(B./(K*u))); % 2(Nd^2) cost
   u=bsxfun(@times,u,exp(log(v)*weights))./v;
end
a=mean(v,2);
```

[BCCNP'15]

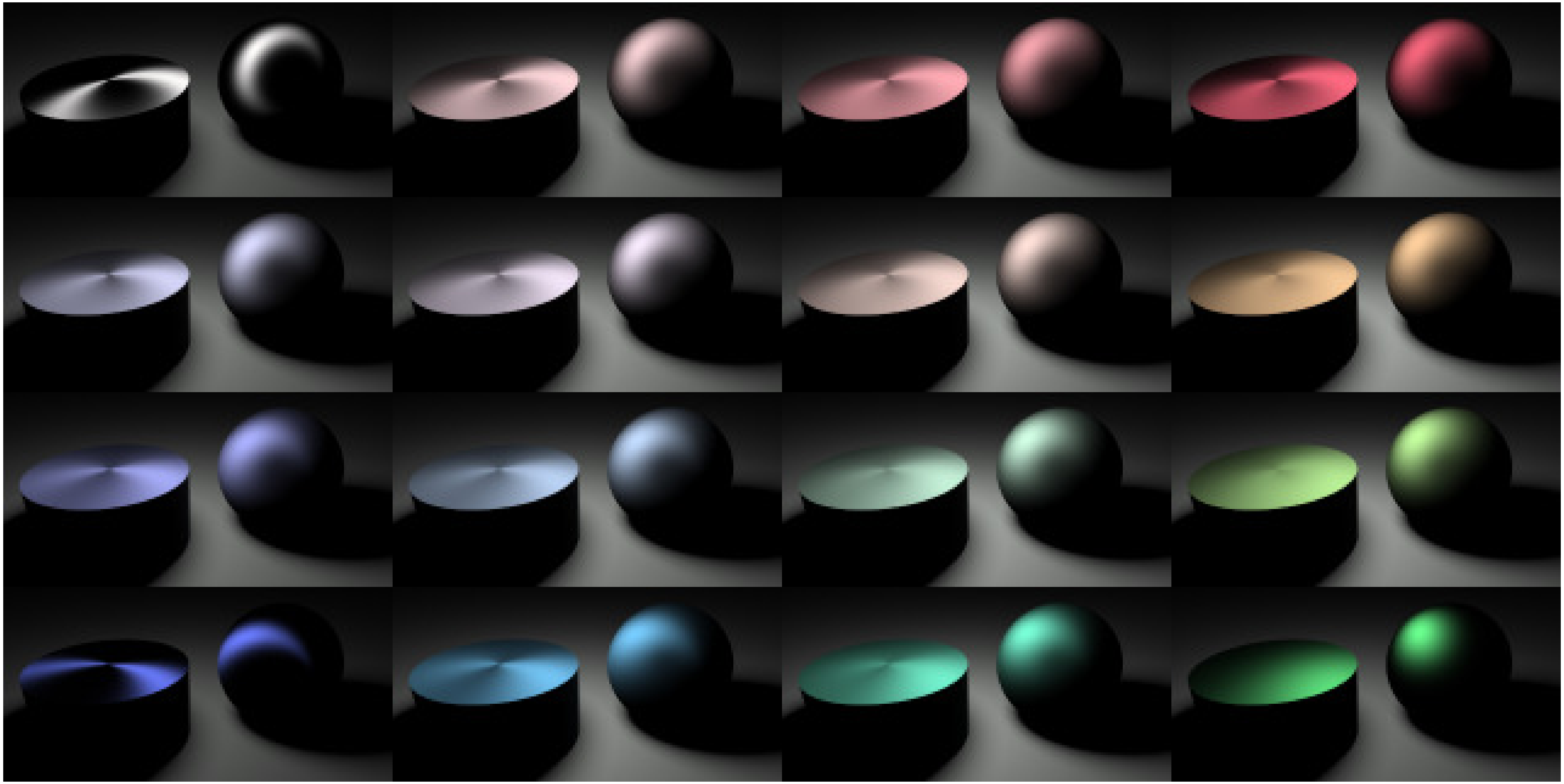
Iterative Bregman Projections for Regularized Transportation Problems
SIAM J. on Sci. Comp. 2015

Applications in Imaging



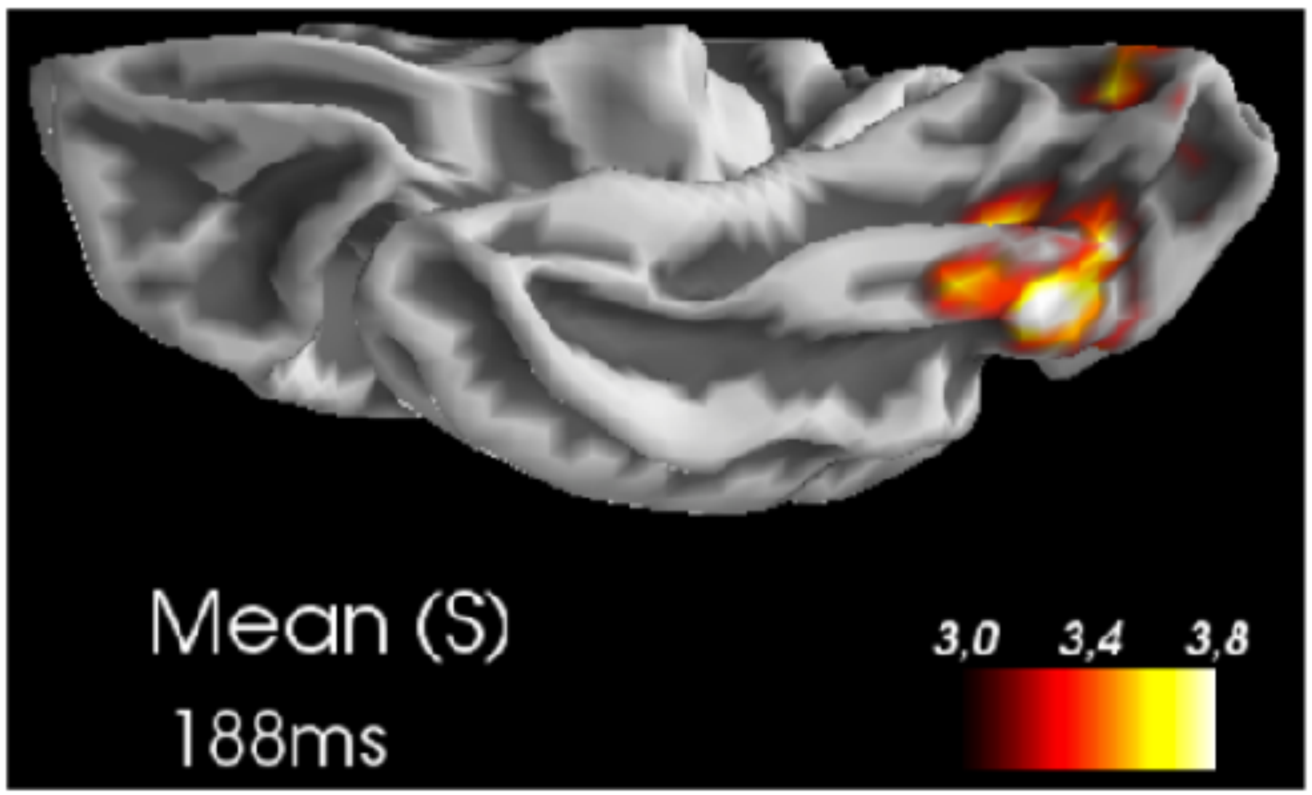
[Solomon'15]

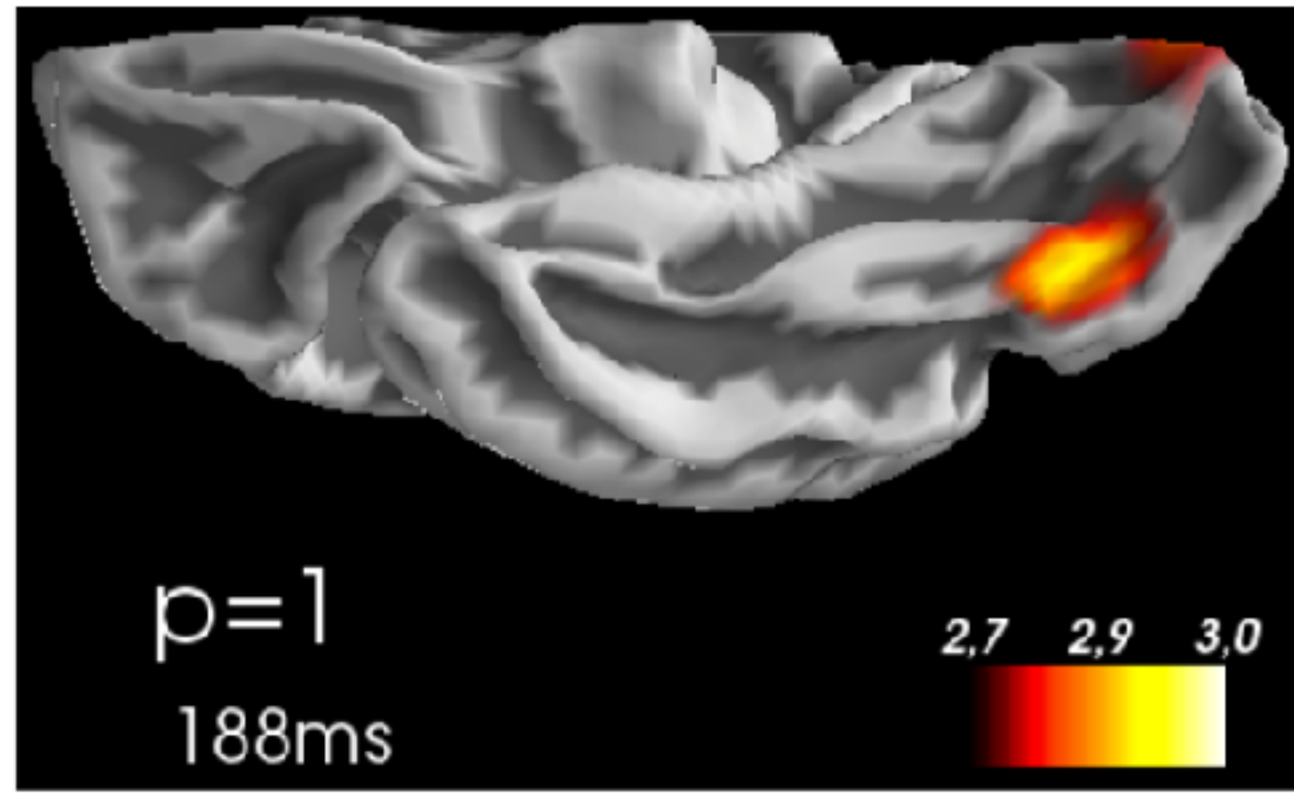
Applications in Imaging



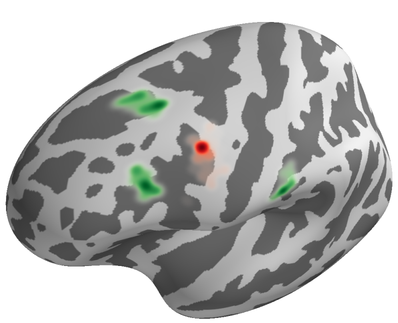
[Solomon'15]

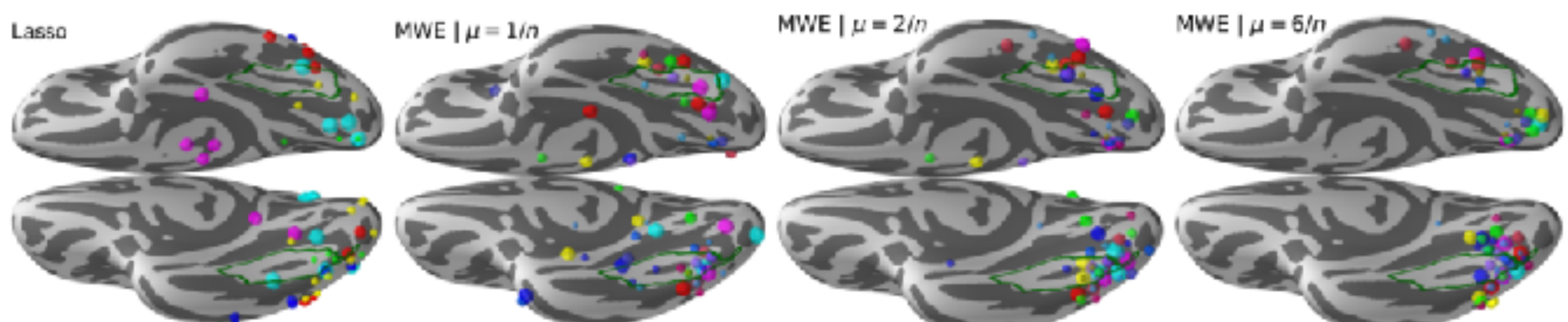
Applications: Brain Imaging



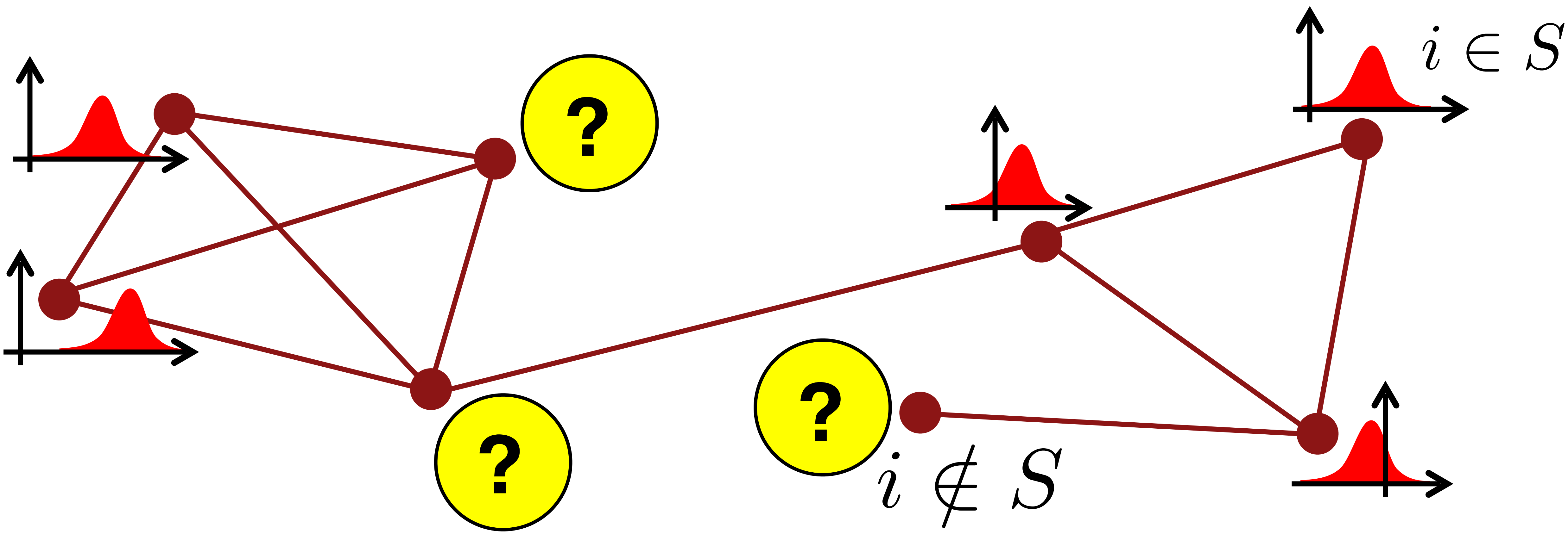


[Gramfort+'14][Janati'+19a,b]





Wasserstein Propagation

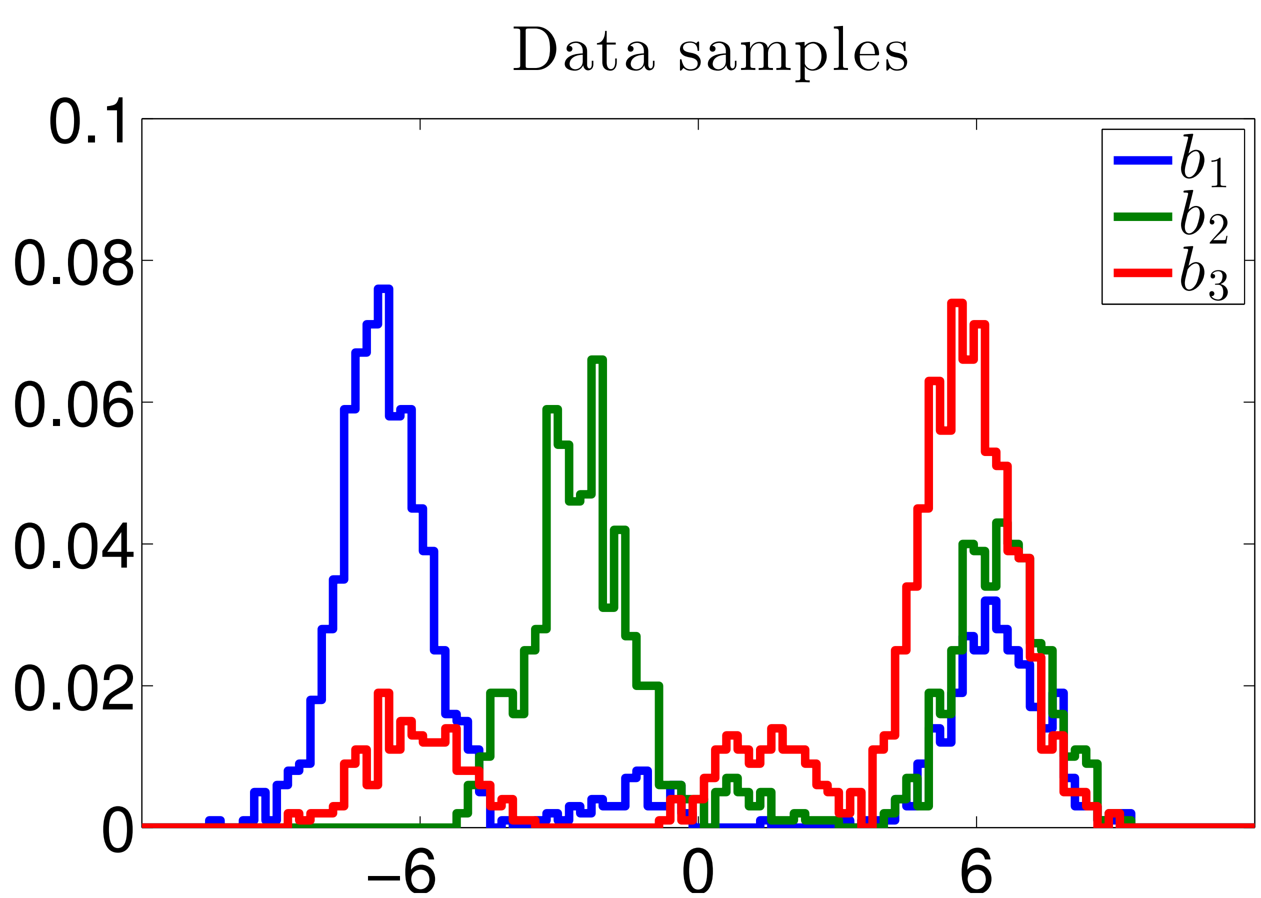


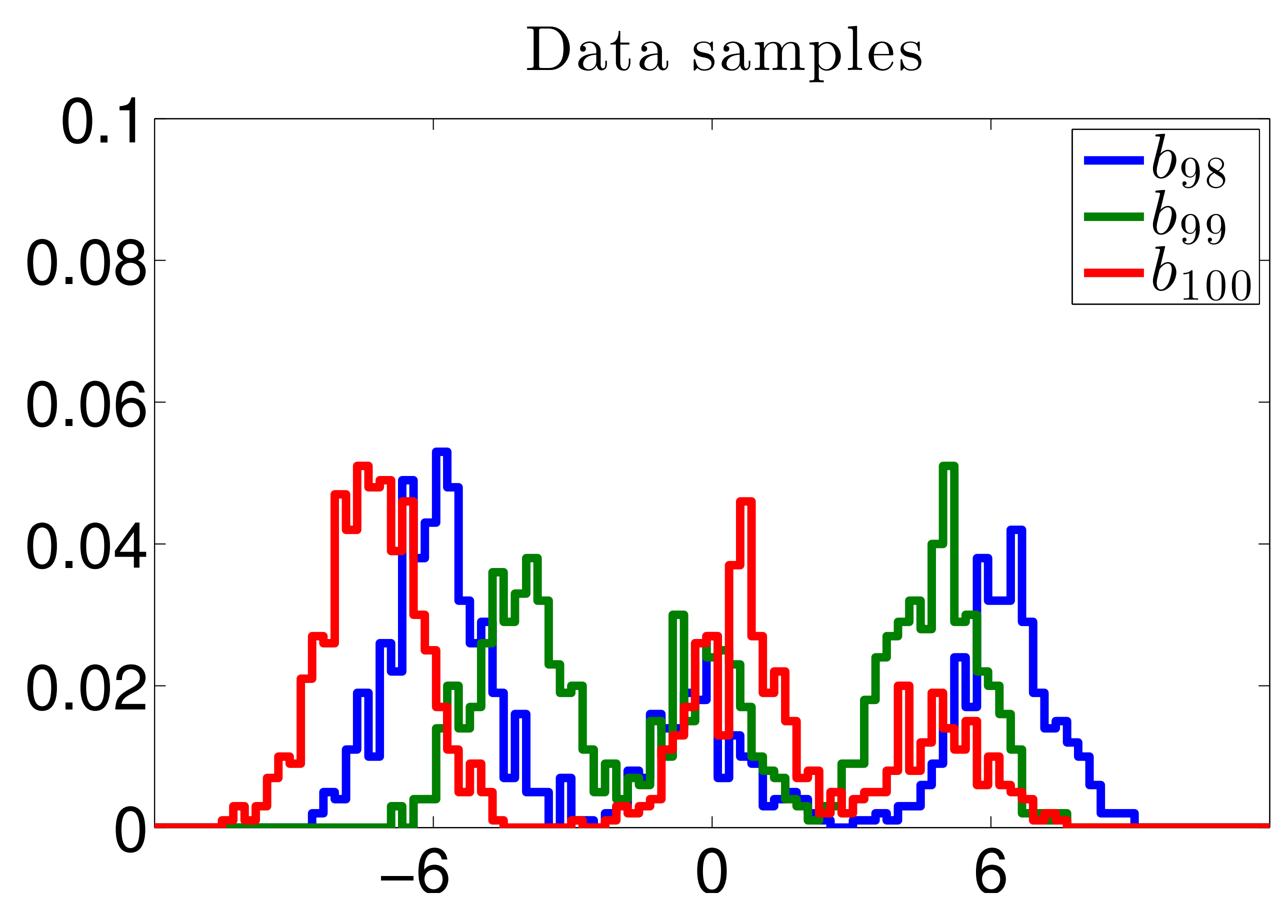
$$\min_{\substack{\mu_i \in \mathcal{P}(\Omega) \\ \mu_i \text{ fixed for } i \in S}} \sum_{(e_1, e_2) \in E} W_2^2(\mu_{e_1}, \mu_{e_2})$$

[Solomon'14]

Dictionary Learning

$$\min_{\boldsymbol{A}\in(\Sigma_n)^K,\boldsymbol{\Lambda}\in(\Sigma_K)^N} \sum_{i=1}^N W\left(\boldsymbol{b_i}, \sum_{k=1}^K \boldsymbol{\Lambda_k^i a_k}\right)$$

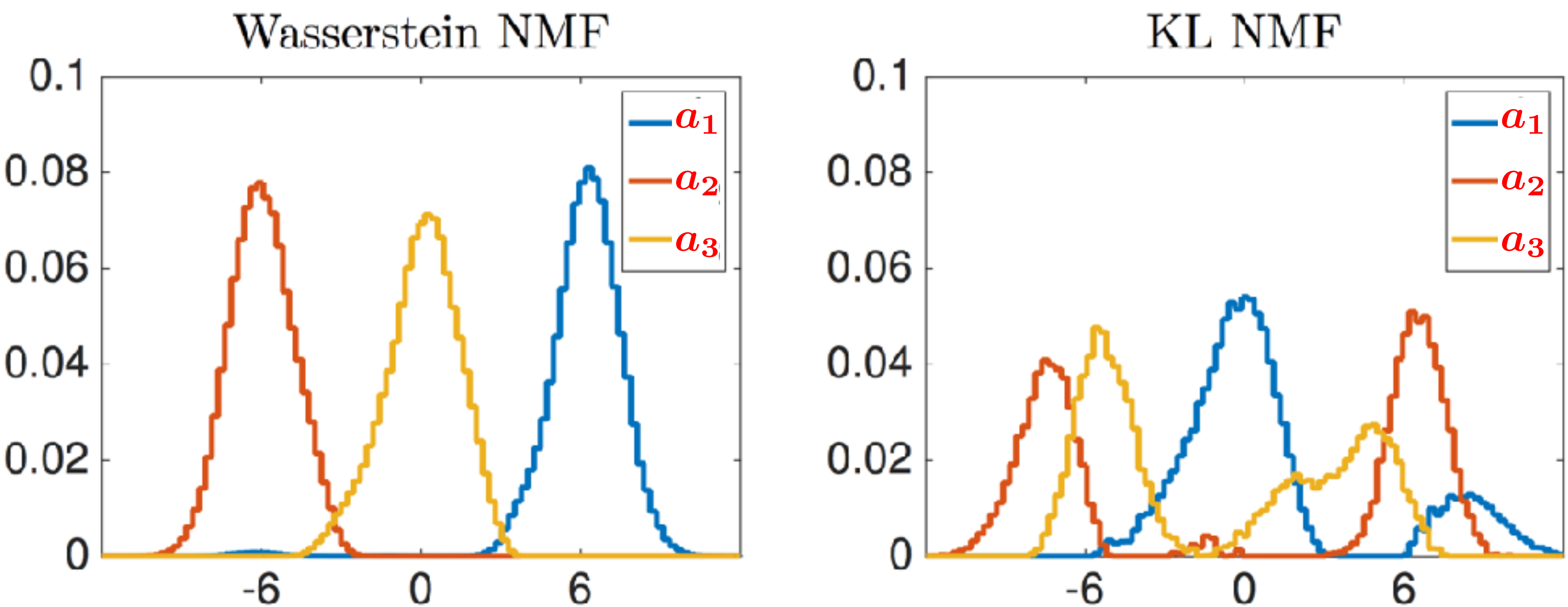




[Sandler'11] [Zen'14] [Rolet'16]

Dictionary Learning

$$\min_{\boldsymbol{A}\in(\Sigma_n)^K,\boldsymbol{\Lambda}\in(\Sigma_K)^N} \sum_{i=1}^N W\left(\boldsymbol{b_i}, \sum_{k=1}^K \boldsymbol{\Lambda}_{\boldsymbol{k}}^{\boldsymbol{i}} \boldsymbol{a_k}\right)$$



[Sandler'11] [Zen'14] [Rolet'16]

OT Dictionary Learning

- **[Hoffman'98]** proposed to learn dictionaries (topics) for text, seen as histograms-of-words.

$$\Omega = \{\text{words}\}, \quad |\Omega| \approx 13,000$$

- Vector embeddings for words **[Mikolov'13]** **[Pennington'14]** defines geometry:

$$\boldsymbol{D}(\text{public}, \text{car}) = \|x_{\text{public}} - x_{\text{car}}\|^2$$

- Data: 7,034 Reuters, 737 BBC sports news articles

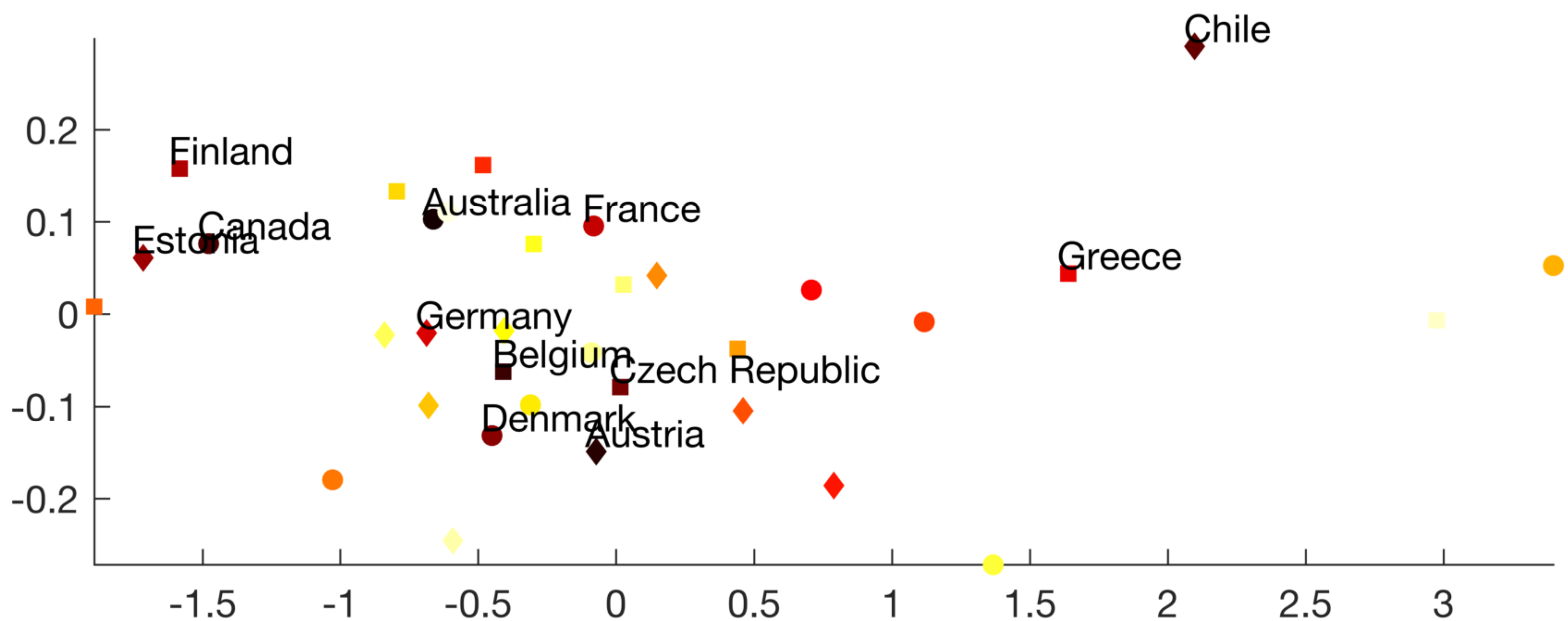
Topic Models

[Rolet'16]

Elliptical Embeddings

Multidimensional Scaling [MDS]

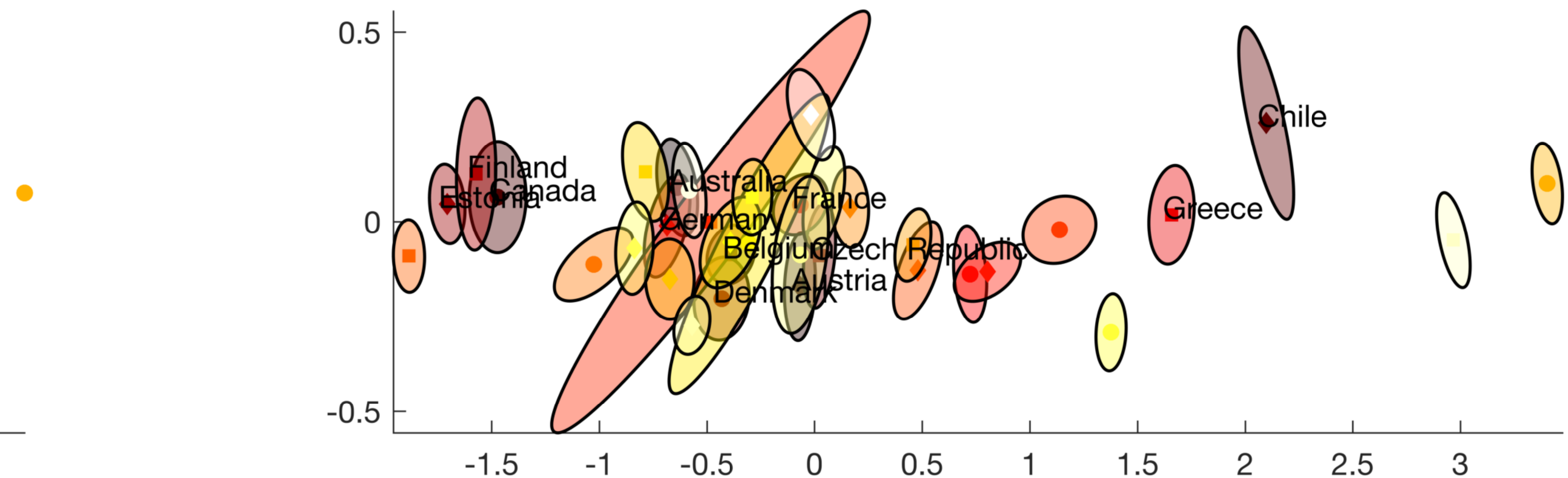
embed a metric space in R^2



Elliptical Embeddings

Multidimensional Scaling [MDS]

embed a metric space in elliptical distributions in $P(R^2)$, W_2



Recall

Remark. If $\Omega = \mathbb{R}^d$, $c(x,y) = \|x-y\|^2$, and $\boldsymbol{\mu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\mu}})$, $\boldsymbol{\nu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\nu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})$ then

$$W_2^2(\boldsymbol{\mu}, \boldsymbol{\nu}) = \|\mathbf{m}_{\boldsymbol{\mu}} - \mathbf{m}_{\boldsymbol{\nu}}\|^2 + B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2$$

where B is the Bures metric

$$B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2 = \operatorname{trace}(\boldsymbol{\Sigma}_{\boldsymbol{\mu}} + \boldsymbol{\Sigma}_{\boldsymbol{\nu}} - 2(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2})^{1/2}).$$

Recall

Remark. If $\Omega = \mathbb{R}^d$, $\boldsymbol{c}(x, y) = \|x - y\|^2$, and $\boldsymbol{\mu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\mu}})$, $\boldsymbol{\nu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\nu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})$ then

$$W_2^2(\boldsymbol{\mu}, \boldsymbol{\nu}) = \|\mathbf{m}_{\boldsymbol{\mu}} - \mathbf{m}_{\boldsymbol{\nu}}\|^2 + B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2$$

where B is the Bures metric

$$B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2 = \mathrm{trace}(\boldsymbol{\Sigma}_{\boldsymbol{\mu}} + \boldsymbol{\Sigma}_{\boldsymbol{\nu}} - 2(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2})^{1/2}).$$

The map $T : x \mapsto \mathbf{m}_{\boldsymbol{\nu}} + A(x - \mathbf{m}_{\boldsymbol{\mu}})$ is **optimal**, where $A = \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-\frac{1}{2}} \left(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{\frac{1}{2}} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{\frac{1}{2}} \right)^{\frac{1}{2}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-\frac{1}{2}}$.

Recall

Remark. If $\Omega = \mathbb{R}^d$, $c(x,y) = \|x-y\|^2$, and $\boldsymbol{\mu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\mu}})$, $\boldsymbol{\nu} = \mathcal{N}(\mathbf{m}_{\boldsymbol{\nu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})$ then

$$W_2^2(\boldsymbol{\mu},\boldsymbol{\nu}) = \|\mathbf{m}_{\boldsymbol{\mu}} - \mathbf{m}_{\boldsymbol{\nu}}\|^2 + B(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})^2$$

where B is the Bures

if $X \sim \mathcal{N}(m, \boldsymbol{\Sigma})$ then $Y := CX + b \sim \mathcal{N}(Cm + b, C\boldsymbol{\Sigma}C^T)$.

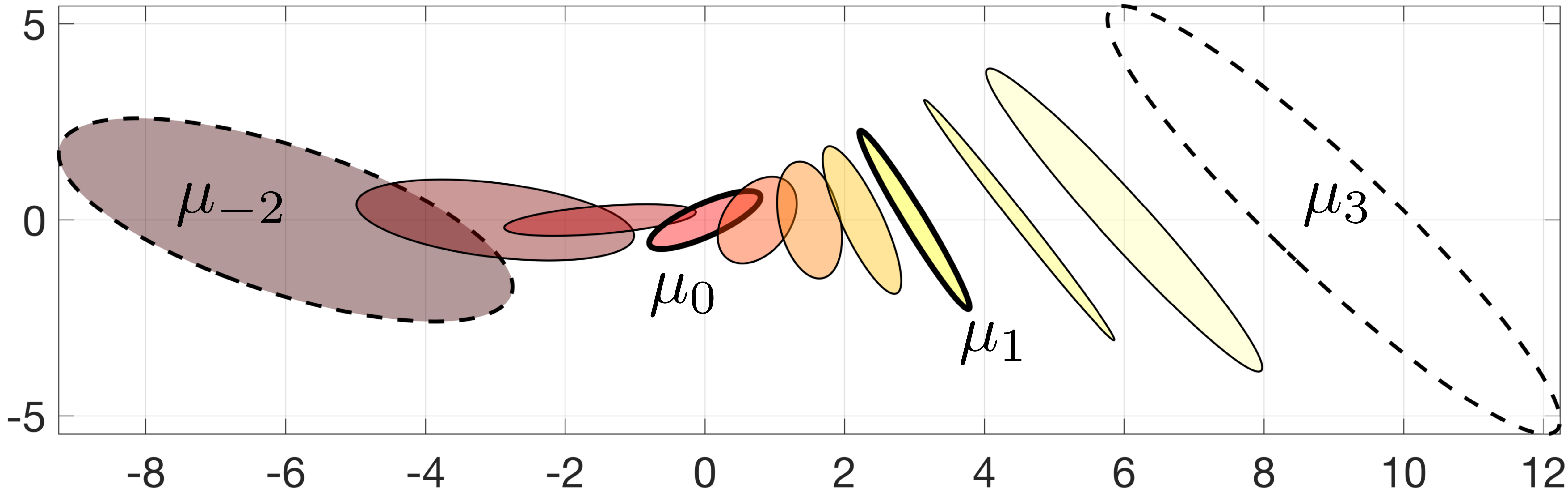
$$\mathrm{trace}(\boldsymbol{\Sigma}_{\boldsymbol{\mu}} + \boldsymbol{\Sigma}_{\boldsymbol{\nu}} - 2(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2}\boldsymbol{\Sigma}_{\boldsymbol{\nu}}\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{1/2})^{1/2}).$$

The map $T : x \mapsto \mathbf{m}_{\boldsymbol{\nu}} + A(x - \mathbf{m}_{\boldsymbol{\mu}})$ is **optimal**,

where $A = \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-\frac{1}{2}} \left(\boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{\frac{1}{2}} \boldsymbol{\Sigma}_{\boldsymbol{\nu}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{\frac{1}{2}}\right)^{\frac{1}{2}} \boldsymbol{\Sigma}_{\boldsymbol{\mu}}^{-\frac{1}{2}}$.

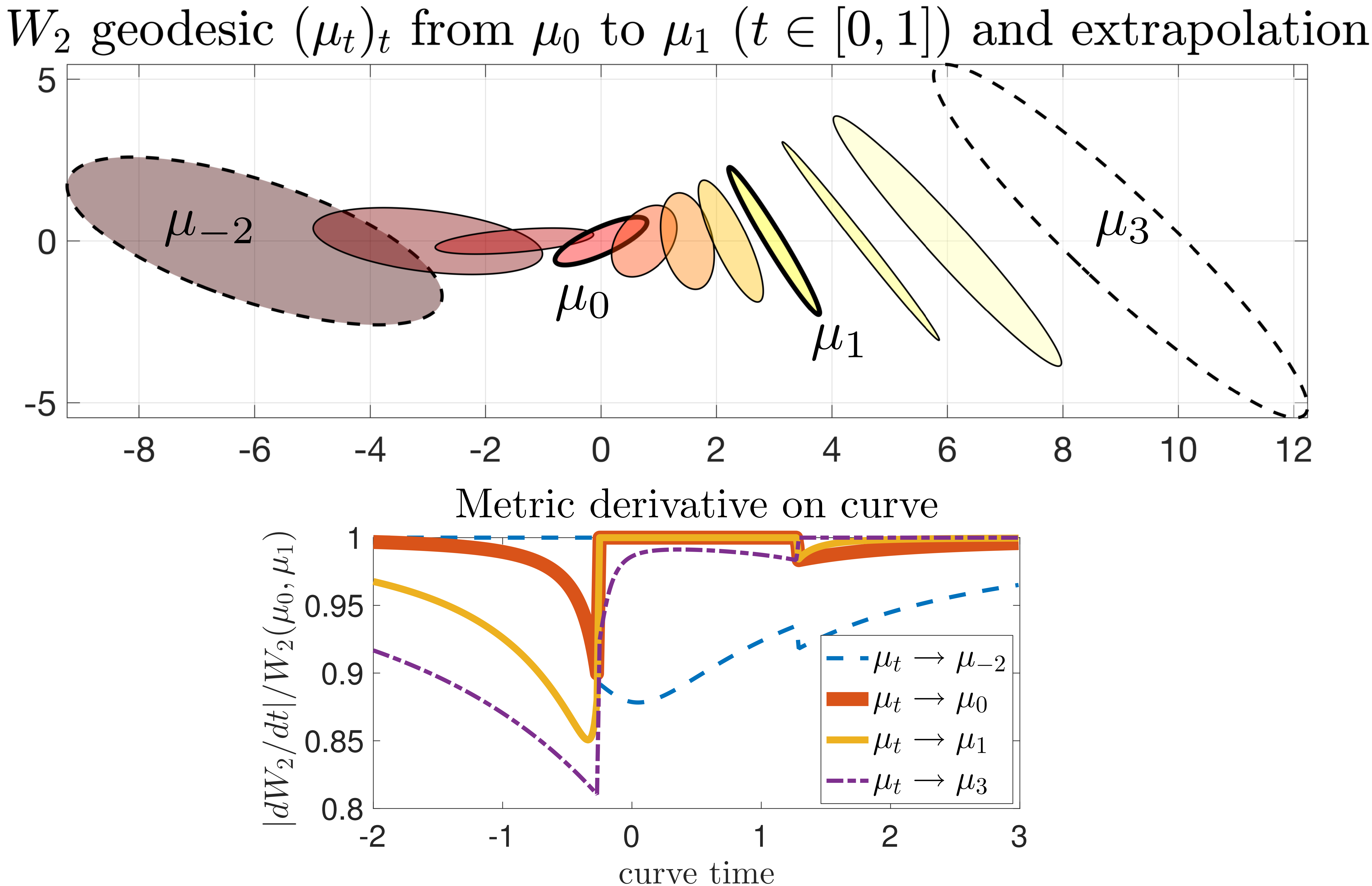
Recall

W_2 geodesic $(\mu_t)_t$ from μ_0 to μ_1 ($t \in [0,1]$) and extrapolation



$$\Sigma_{\boldsymbol{t}} = ((1-t)\,I + tA)\,\Sigma_{\boldsymbol{\mu}}\,((1-t)\,I + tA)$$

Recall


W_2 geodesic $(\mu_t)_t$ from μ_0 to μ_1 ($t \in [0,1]$) and extrapolation
5
0
-5
-8
-6
-4
-2
0
2
4
6
8
10
12
μ_{-2}
μ_0
μ_1
μ_3
Metric derivative on curve
$|dW_2/dt|/W_2(\mu_0,\mu_1)$
1
0.95
0.9
0.85
0.8
-2
-1
0
1
2
3
curve time
$\mu_t \to \mu_{-2}$
$\mu_t \to \mu_0$
$\mu_t \to \mu_1$
$\mu_t \to \mu_3$

Computations

$$\mathbf{A} = \mathbf{L}\mathbf{L}^T$$

$$\nabla_{\mathbf{L}} \frac{1}{2} \mathfrak{B}^2(\mathbf{A}, \mathbf{B}) = \left(\mathbf{I} - \mathbf{T}^{\mathbf{AB}}\right) \mathbf{L}$$

$$\mathbf{L}' = \left((1-\eta)\mathbf{I} + \eta \mathbf{T}^{\mathbf{AB}}\right) \mathbf{L}$$

Algorithm 1 Newton-Schulz

Input: PSD matrix $\mathbf{A}$, $\epsilon > 0$

$\mathbf{Y} \leftarrow \frac{\mathbf{A}}{(1+\epsilon)\|\mathbf{A}\|}$, $\mathbf{Z} \leftarrow \mathbf{I}$

while not converged **do**

$\mathbf{T} \leftarrow (3\mathbf{I} - \mathbf{ZY})/2$

$\mathbf{Y} \leftarrow \mathbf{YT}$

$\mathbf{Z} \leftarrow \mathbf{TZ}$

end while

$\mathbf{Y} \leftarrow \sqrt{(1+\epsilon)\|\mathbf{A}\|}\mathbf{Y}$

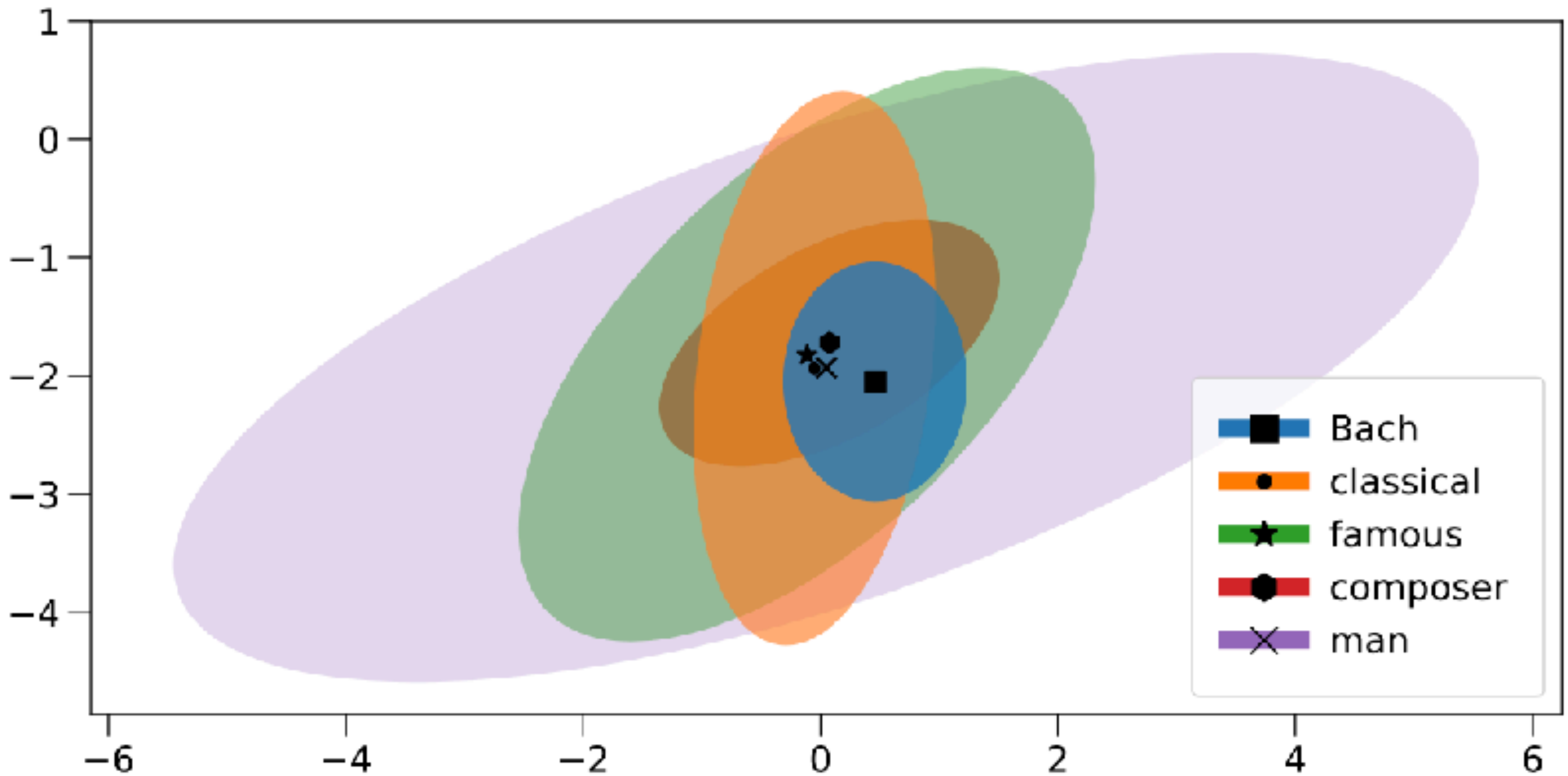
$\mathbf{Z} \leftarrow \frac{\mathbf{Z}}{\sqrt{(1+\epsilon)\|\mathbf{A}\|}}$

Output: square root $\mathbf{Y}$, inverse square root $\mathbf{Z}$

Elliptical Embeddings

Word Embeddings

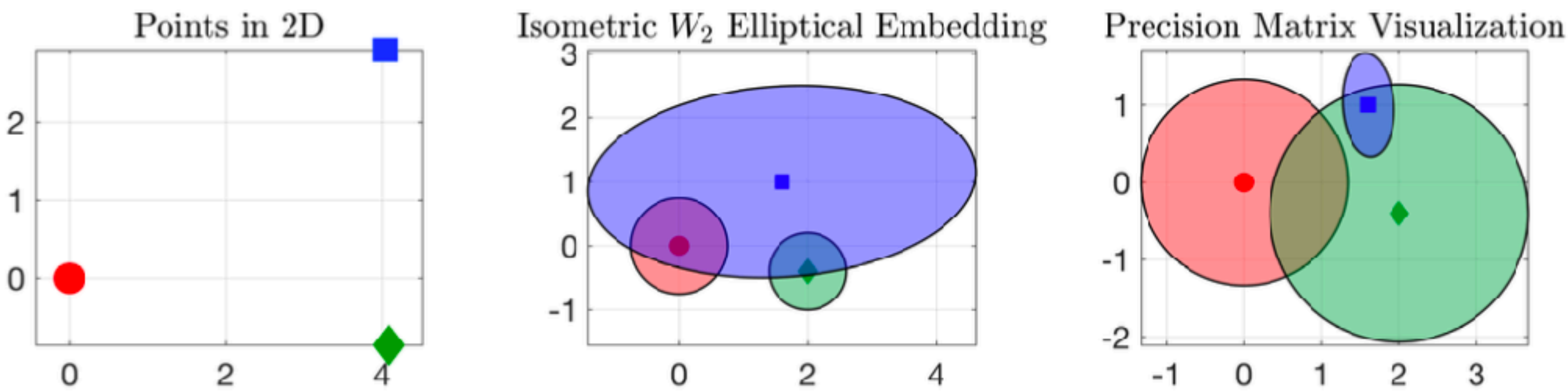
Compute elliptical distribution representations for Words



Elliptical Embeddings

Visualization issue

need to shift to precision matrix to recover intuition



Inverse Wasserstein Problems

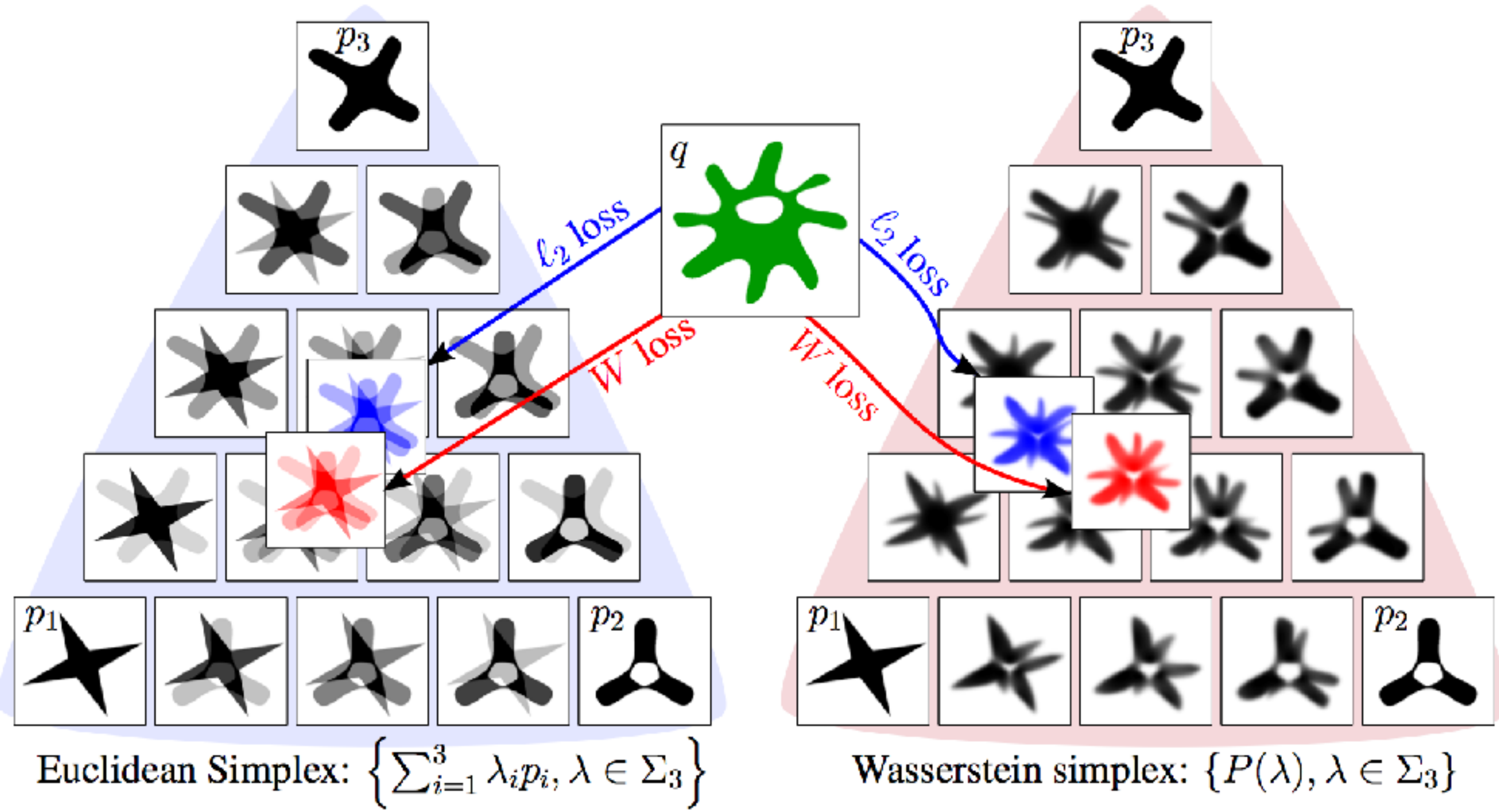
- consider Barycenter operator:

$$\boldsymbol{b}(\lambda) \stackrel{\text{def}}{=} \underset{\boldsymbol{a}}{\operatorname{argmin}} \sum_{i=1}^{N} \lambda_i W_\gamma(\boldsymbol{a}, \boldsymbol{b_i})$$

- address now **Wasserstein inverse problems**:

$$\text{Given } \boldsymbol{a}, \text{ find } \underset{\lambda \in \Sigma_N}{\operatorname{argmin}} \mathcal{E}(\lambda) \stackrel{\text{def}}{=} \operatorname{Loss}(\boldsymbol{a}, \boldsymbol{b}(\lambda))$$

Wasserstein Inverse Problems



Barycenters = Fixed Points

Prop. **[BCCNP'15]** Consider $\boldsymbol{B} \in \Sigma_d^N$ and let $\boldsymbol{U_0} = \mathbf{1}_{\boldsymbol{d\times N}}$, and then for $l \geq 0$:

$$\boldsymbol{b}^l \stackrel{\text{def}}{=} \exp\left(\log\left(K^T \boldsymbol{U_l}\right)\lambda\right); \begin{cases} \boldsymbol{V_{l+1}} \stackrel{\text{def}}{=} \dfrac{\boldsymbol{b}^l \mathbf{1}_N^T}{K^T \boldsymbol{U_l}}, \\ \boldsymbol{U_{l+1}} \stackrel{\text{def}}{=} \dfrac{\boldsymbol{B}}{K \boldsymbol{V_{l+1}}}. \end{cases}$$

Using Truncated Barycenters

- instead of using the exact barycenter

$$\operatorname*{argmin}_{\lambda \in \Sigma_N} \mathcal{E}(\lambda) \stackrel{\text{def}}{=} \text{Loss}(\boldsymbol{a}, \boldsymbol{b}(\lambda))$$

- use instead the L-iterate barycenter

$$\operatorname*{argmin}_{\lambda \in \Sigma_N} \mathcal{E}^{(L)}(\lambda) \stackrel{\text{def}}{=} \text{Loss}(\boldsymbol{a}, \boldsymbol{b}^{(L)}(\lambda))$$

- Differente using **the chain rule.**

$$\nabla \mathcal{E}^{(L)}(\lambda) = [\partial \boldsymbol{b}^{(L)}]^T(\boldsymbol{g}), \;\; \boldsymbol{g} \stackrel{\text{def}}{=} \nabla \text{Loss}(\boldsymbol{a}, \cdot)|_{\boldsymbol{b}^{(L)}(\lambda)}.$$

Gradient / Barycenter Computation

function SINKHORN-DIFFERENTIATE($(p_s)_{s=1}^S, q, \lambda$)

$\forall s, b_s^{(0)} \leftarrow \mathbb{1}$

$(w, r) \leftarrow (0^S, 0^{S\times N})$

for $\ell = 1, 2, \dots, L$ *// Sinkhorn loop*

$\forall s, \varphi_s^{(\ell)} \leftarrow K^\top \frac{p_s}{K b_s^{(\ell-1)}}$

$p \leftarrow \prod_s \left(\varphi_s^{(\ell)}\right)^{\lambda_s}$

$\forall s, b_s^{(\ell)} \leftarrow \frac{p}{\varphi_s^{(\ell)}}$

$g \leftarrow \nabla \mathcal{L}(p, q) \odot p$

for $\ell = L, L-1, \dots, 1$ *// Reverse loop*

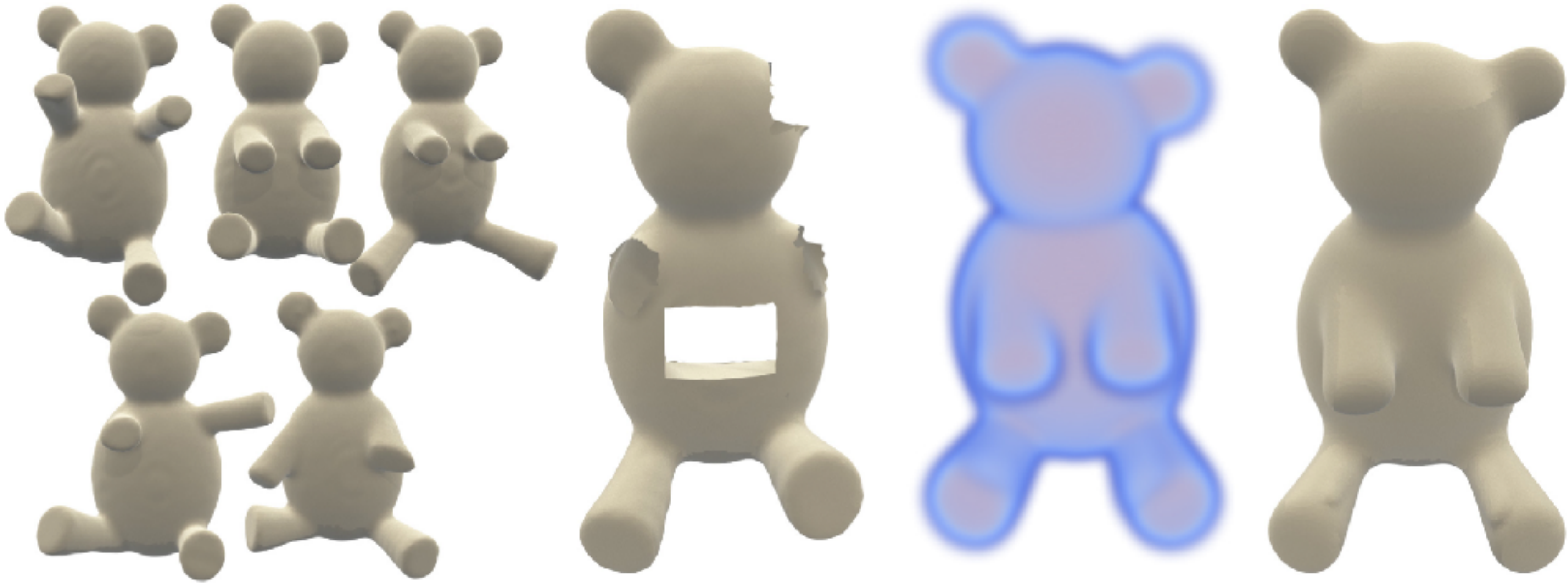
$\forall s, w_s \leftarrow w_s + \langle \log \varphi_s^{(\ell)}, g \rangle$

$\forall s, r_s \leftarrow -K^\top (K(\frac{\lambda_s g - r_s}{\varphi_s^{(\ell)}}) \odot \frac{p_s}{(K b_s^{(\ell-1)})^2}) \odot b_s^{(\ell-1)}$

$g \leftarrow \sum_s r_s$

return $P^{(L)}(\lambda) \leftarrow p, \nabla \mathcal{E}_L(\lambda) \leftarrow w$

Application: Volume Reconstruction



Shape database $(p_1, \ldots, p_5)$ | Input shape q | Projection $P(\lambda)$ | Iso-surface

[Bonneel'16]

Application: Color Grading

Application: Color Grading

$\lambda_0 = 0.03$

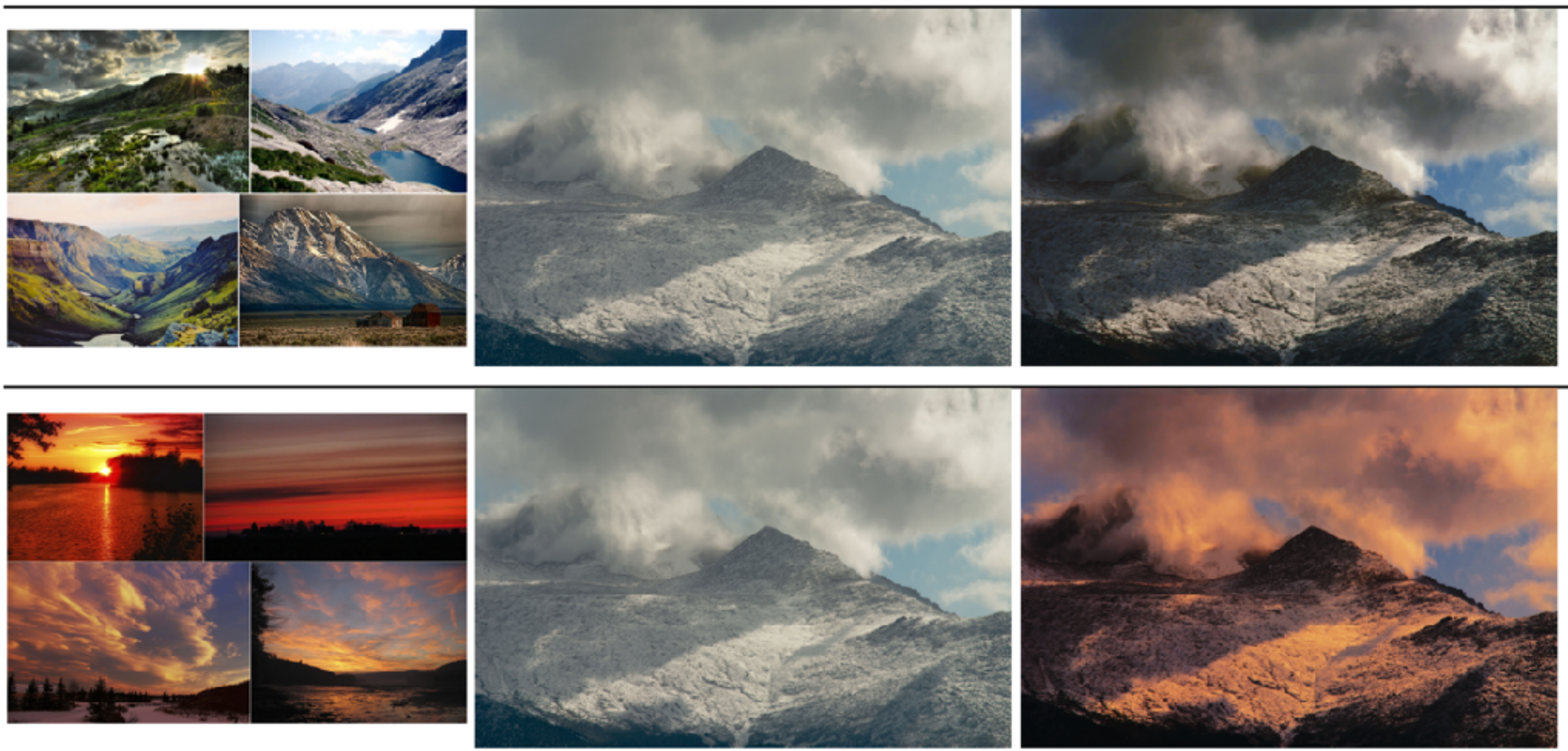
$\lambda_1 = 0.12$

$\lambda_2 = 0.40$

$\lambda_3 = 0.43$

Application: Color Grading

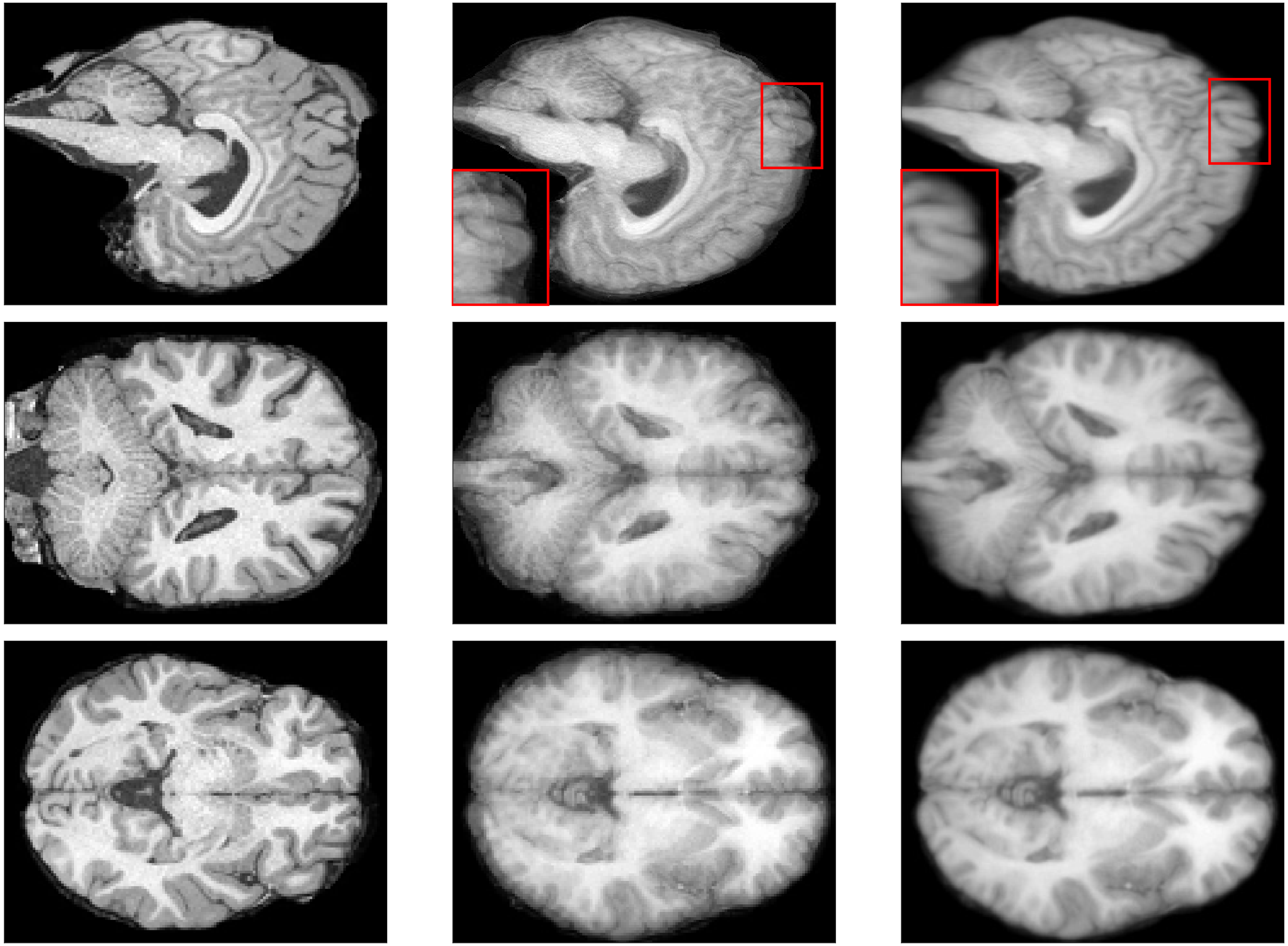
Application: Color Grading



Wasserstein Barycentric Coordinates: Histogram Regression using Optimal Transport, **SIGGRAPH'16**

[BPC'16]

Application: Brain Mapping

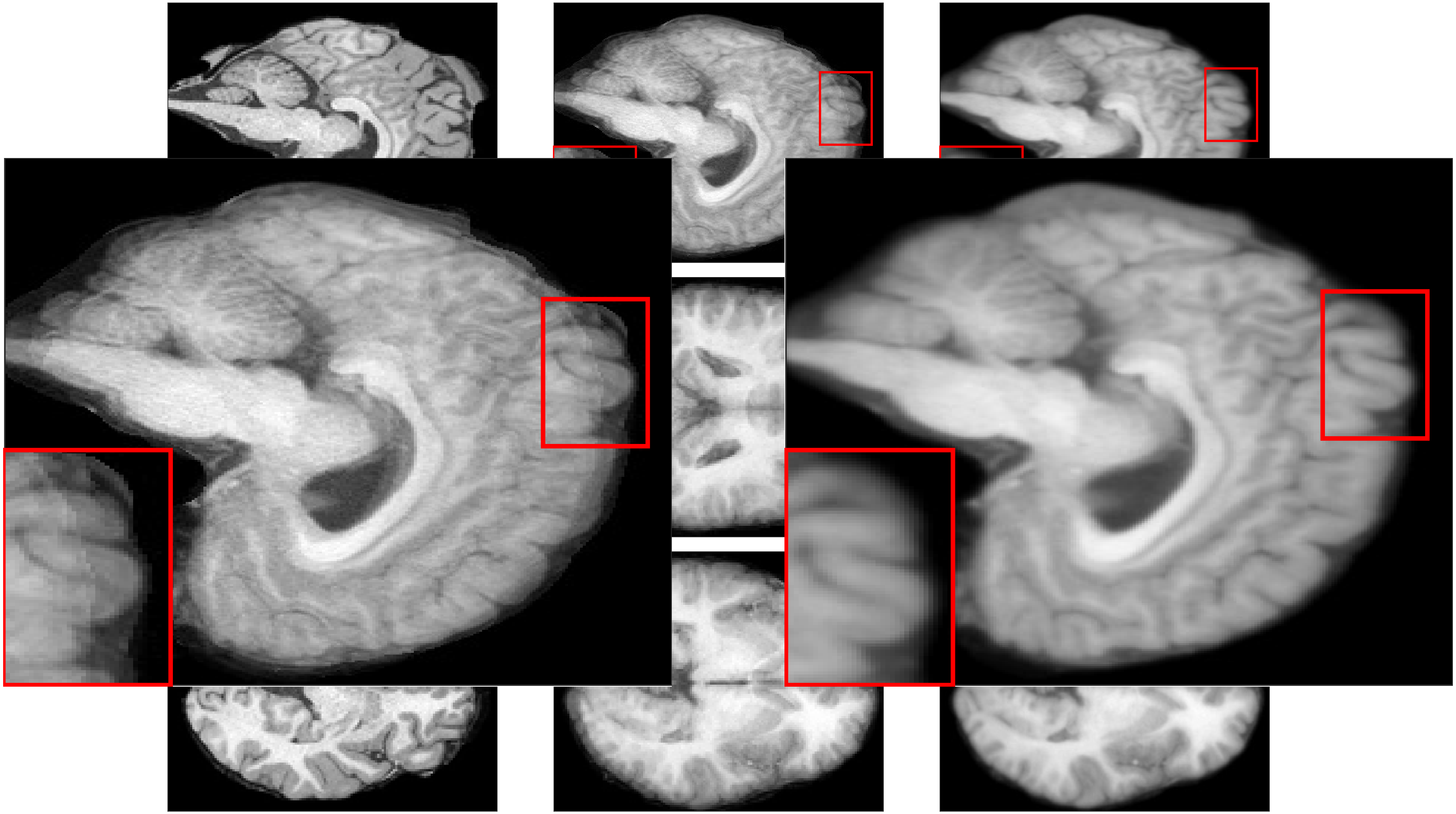


Original

Euclidean projection

Wasserstein projection

Application: Brain Mapping



Original

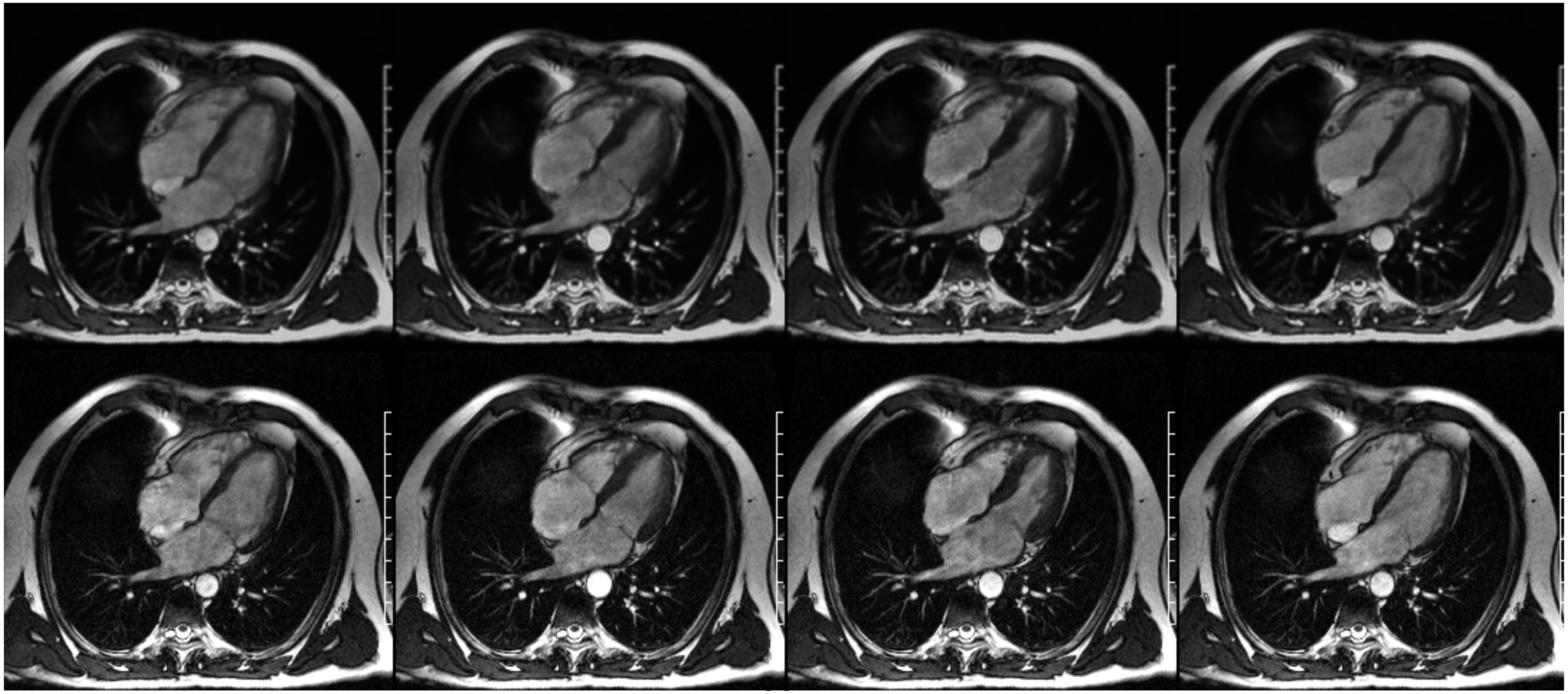
Euclidean projection

Wasserstein projection

end-to-end W Dictionary Learning

$$\min_{\boldsymbol{A}\in(\Sigma_n)^K \boldsymbol{\Lambda}\in(\Sigma_K)^N} \sum_{i=1}^{N} \mathcal{L}\left(\boldsymbol{b_i}, \boldsymbol{a}(\boldsymbol{\lambda_i})\right)$$

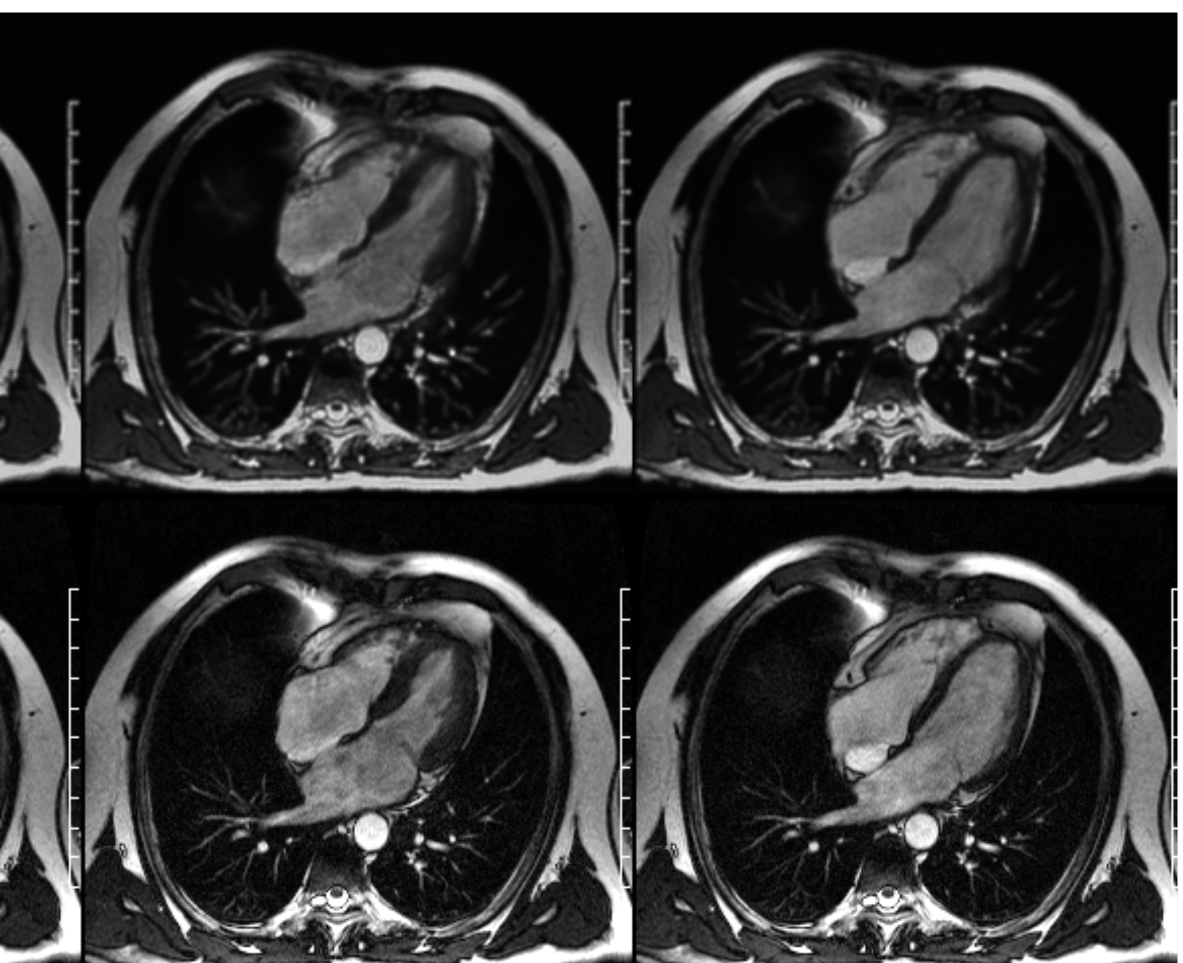
[Schmitz'18]

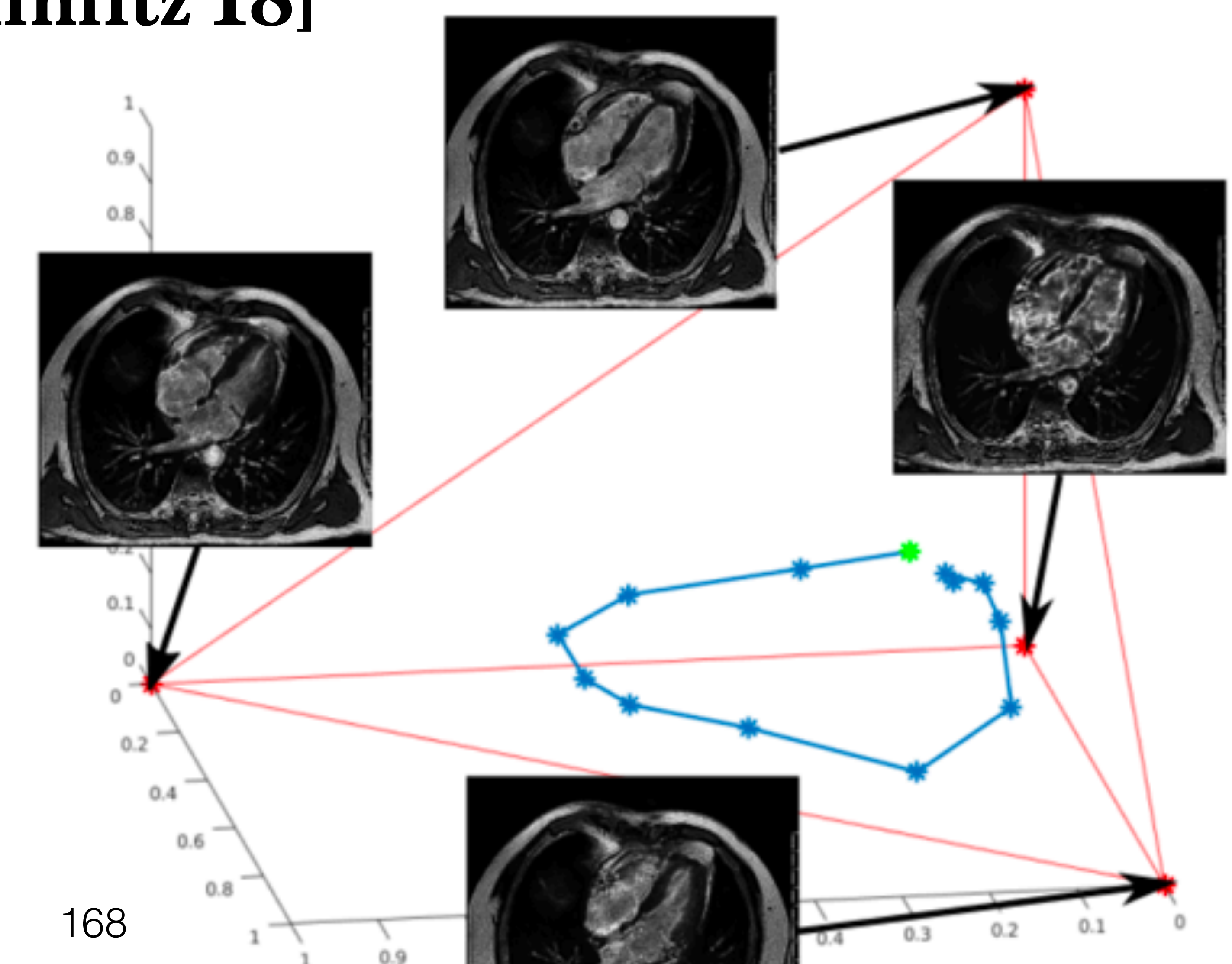


end-to-end W Dictionary Learning

$$\min_{\boldsymbol{A}\in(\Sigma_n)^K \boldsymbol{\Lambda}\in(\Sigma_K)^N} \sum_{i=1}^{N} \mathcal{L}\left(\boldsymbol{b_i}, \boldsymbol{a}(\boldsymbol{\lambda_i})\right)$$

[Schmitz'18]





Distributionally Robust Optimization

$$\boldsymbol{\nu}_{\text{data}} = \frac{\mathbf{1}}{\boldsymbol{n}} \sum_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{N}} \boldsymbol{\delta}_{(\boldsymbol{x_i}, \boldsymbol{y_i})}$$

Supervised learning

$$\inf_{\theta \in \Theta} \mathbb{E}_{\boldsymbol{\nu}_{\text{data}}} [\mathcal{L}(f_\theta(X), Y)]$$

Learning with Wasserstein Ambiguity

$$\inf_{\theta \in \Theta} \sup_{\boldsymbol{\mu} : W_p(\boldsymbol{\nu}_{\text{data}}, \boldsymbol{\mu}) < \varepsilon} \mathbb{E}_{\boldsymbol{\mu}} [\mathcal{L}(f_\theta(X), Y)]$$

[Esvahani'17]

Distributionally Robust Learning

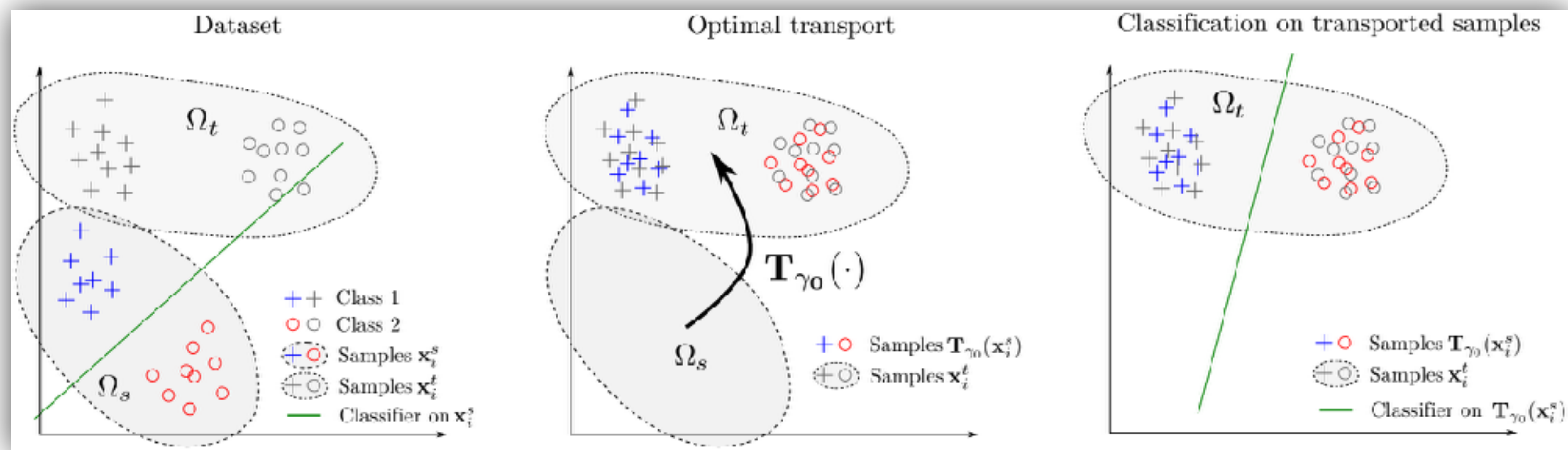
Learning with Wasserstein Ambiguity

$$\inf_{\theta \in \Theta} \sup_{\boldsymbol{\mu} : W_p(\boldsymbol{\nu}_{\text{data}}, \boldsymbol{\mu}) < \varepsilon} \mathbb{E}_{\boldsymbol{\mu}}[\mathcal{L}(f_\theta(X), Y)]$$

Advantages:

- Bound on out-of-sample performance
- Converges as size of dataset increases
- Often reduces to a finite convex program (e.g. when f is element-wise max over elementary concave functions)

Domain Adaptation



1. Estimate transport map
2. Transport labeled samples to new domain
3. Train classifier on transported labeled samples

[Courty'16]

Learning with a Wasserstein Loss

$$\text{Dataset } \{(x_i, y_i)\}, x_i \in \mathbb{R}^p, y_i \in \mathbb{R}^n_+$$

x_i

husky
snow
sled
slope
men

y_i

Goal is to find $f_{\boldsymbol{\theta}}$: Images $\mapsto$ Labels

Learning with a Wasserstein Loss

$$\min_{\boldsymbol{\theta} \in \Theta} \sum_{i=1}^{N} \mathcal{L}(f_{\boldsymbol{\theta}}(x_i), y_i)$$

x_i

husky
snow
sled
slope
men

y_i

Which loss $\mathcal{L}$ could we use?

Learning with a Wasserstein Loss

$$\min_{\boldsymbol{\theta}\in\Theta} \sum_{i=1}^{N} \mathcal{L}(f_{\boldsymbol{\theta}}(x_i), y_i)$$

dog driver winter ice $f_{\boldsymbol{\theta}}(x_i)$	husky snow sled slope men y_i

Which loss $\mathcal{L}$ could we use?

Learning with a Wasserstein Loss

$$\min_{\boldsymbol{\theta} \in \Theta} \sum_{i=1}^{N} \mathcal{L}(f_{\boldsymbol{\theta}}(x_i), y_i)$$

$$\mathcal{L}(\boldsymbol{a}, \boldsymbol{b}) = \min_{\boldsymbol{P} \in \mathbb{R}^{nm}} \langle \boldsymbol{P}, M \rangle + \varepsilon \mathrm{KL}(\boldsymbol{P}\mathbf{1}, \boldsymbol{a}) + \varepsilon \mathrm{KL}(\boldsymbol{P}^T \mathbf{1}, \boldsymbol{b}) - \gamma E(\boldsymbol{P})$$

1. Generalizes Word Mover's to label clouds
2. Sinkhorn algorithm can be generalized

[Frogner'15] [Chizat'15][Chizat'16]

Life Sciences

- Biology to infer evolution of cells

[Hashimoto+'16] […. Rigollet …'19 : *Waddington OT* **]**

Cell

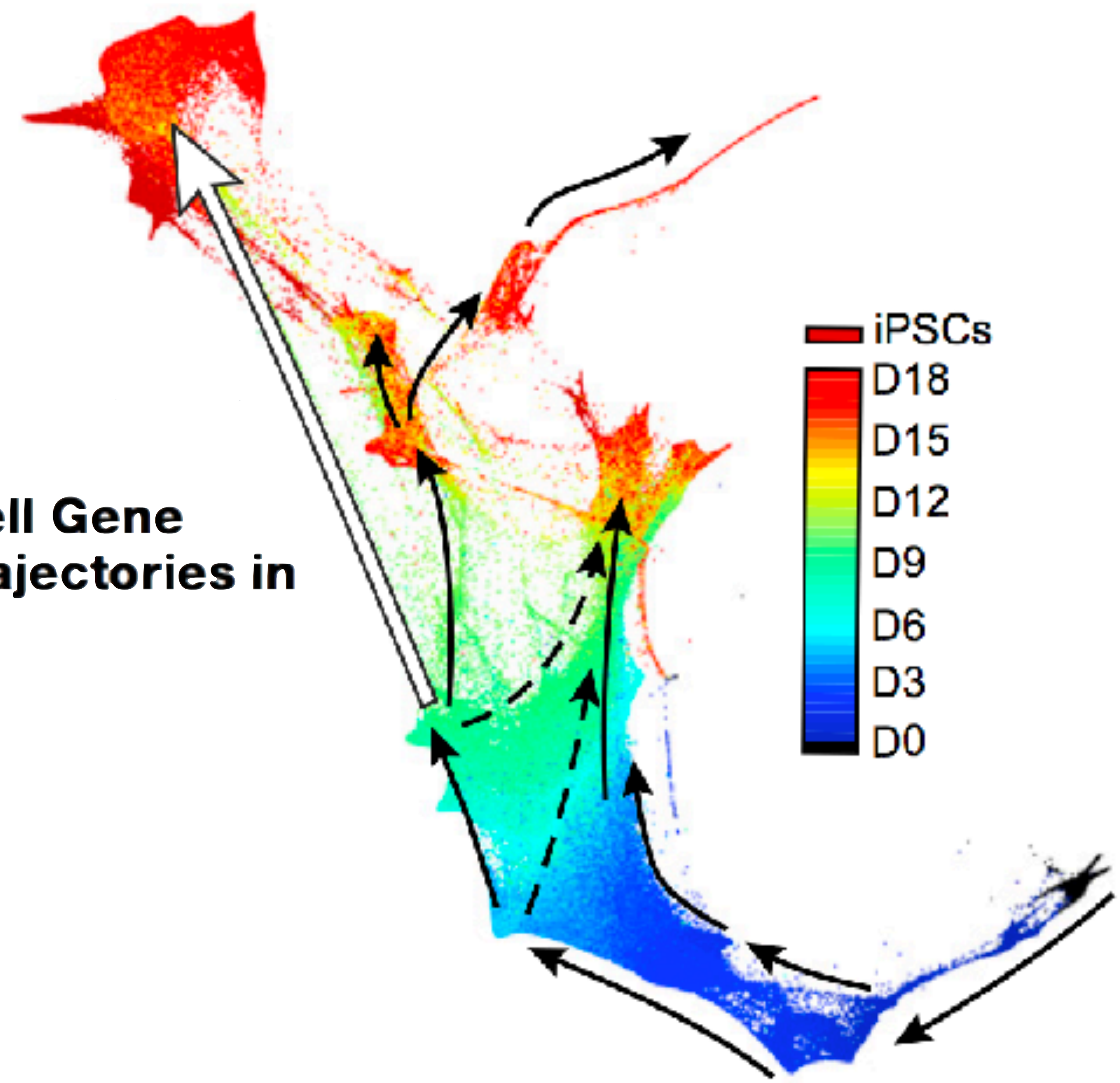
Optimal-Transport Analysis of Single-Cell Gene Expression Identifies Developmental Trajectories in Reprogramming

Life Sciences

- Biology to infer evolution of cells
  [Hashimoto+'16] […. Rigollet …'19 : *Waddington OT* **]**

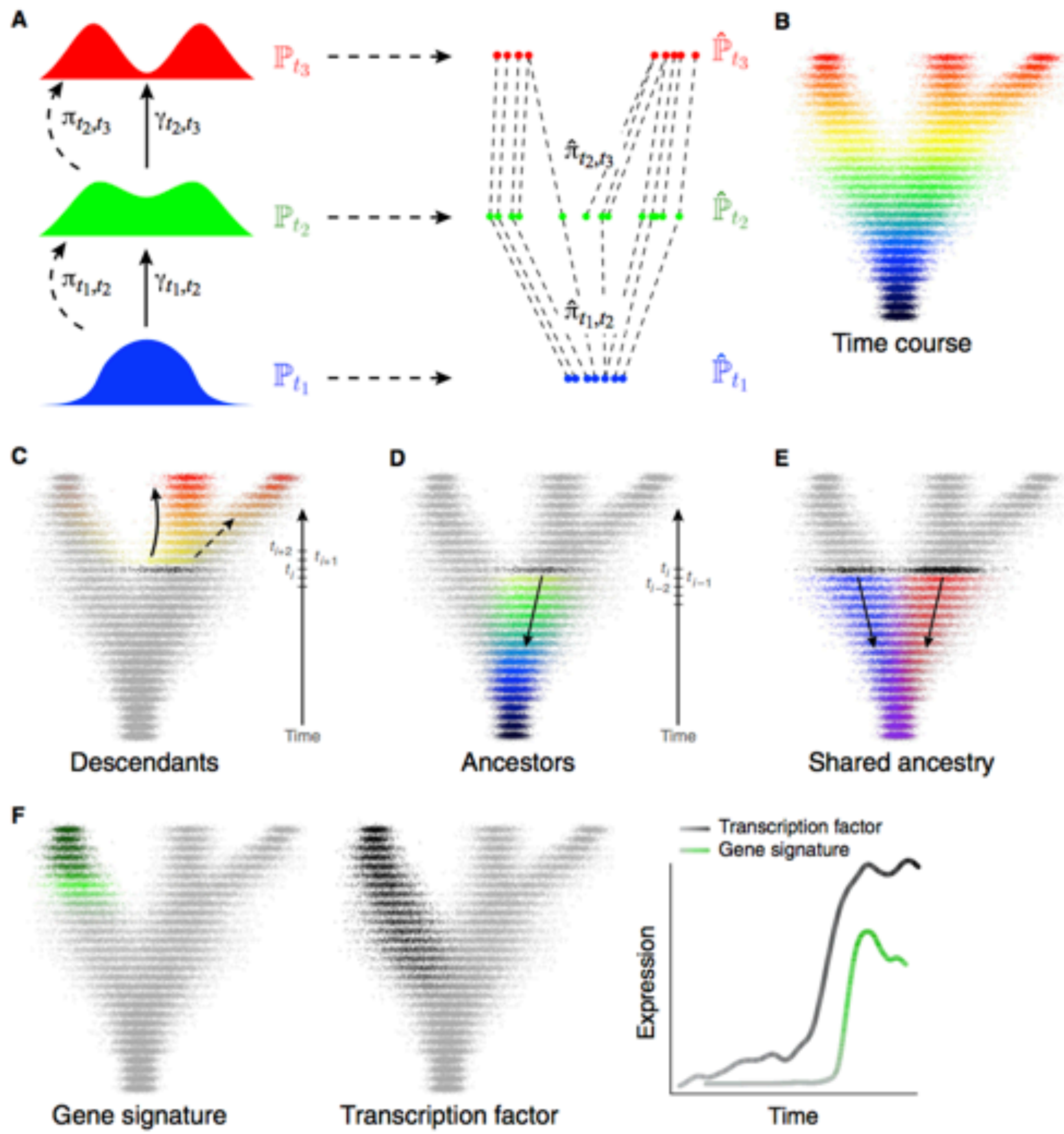
Cell

Optimal-Transport Analysis of Single-Cell Gene Expression Identifies Developmental Trajectories in Reprogramming



Life Sciences

- […. **Rigollet …'19** : *Waddington OT*]



Minimum Kantorovich Estimation

Available online at www.sciencedirect.com

SCIENCE DIRECT

Statistics & Probability Letters 76 (2006) 1298–1302

STATISTICS & PROBABILITY LETTERS

www.elsevier.com/locate/stapro

On minimum Kantorovich distance estimators

Federico Bassetti[a], Antonella Bodini[b], Eugenio Regazzini[a,*]

Use *Wasserstein distances* to define a loss between data and model.

$$\min_{\boldsymbol{\theta} \in \Theta} W(\boldsymbol{\nu}_{\text{data}}, p_{\boldsymbol{\theta}})$$

Minimum Kantorovich Estimators

$$\min_{\boldsymbol{\theta}\in\Theta} W(\boldsymbol{\nu}_{\text{data}}, f_{\boldsymbol{\theta}\sharp}\boldsymbol{\mu})$$

[Bassetti'06] 1st reference discussing this approach.

$$\text{Challenge: } \nabla_{\boldsymbol{\theta}} W(\boldsymbol{\nu}_{\text{data}}, f_{\boldsymbol{\theta}\sharp}\boldsymbol{\mu})?$$

[Montavon'16] use regularized OT in a finite setting.

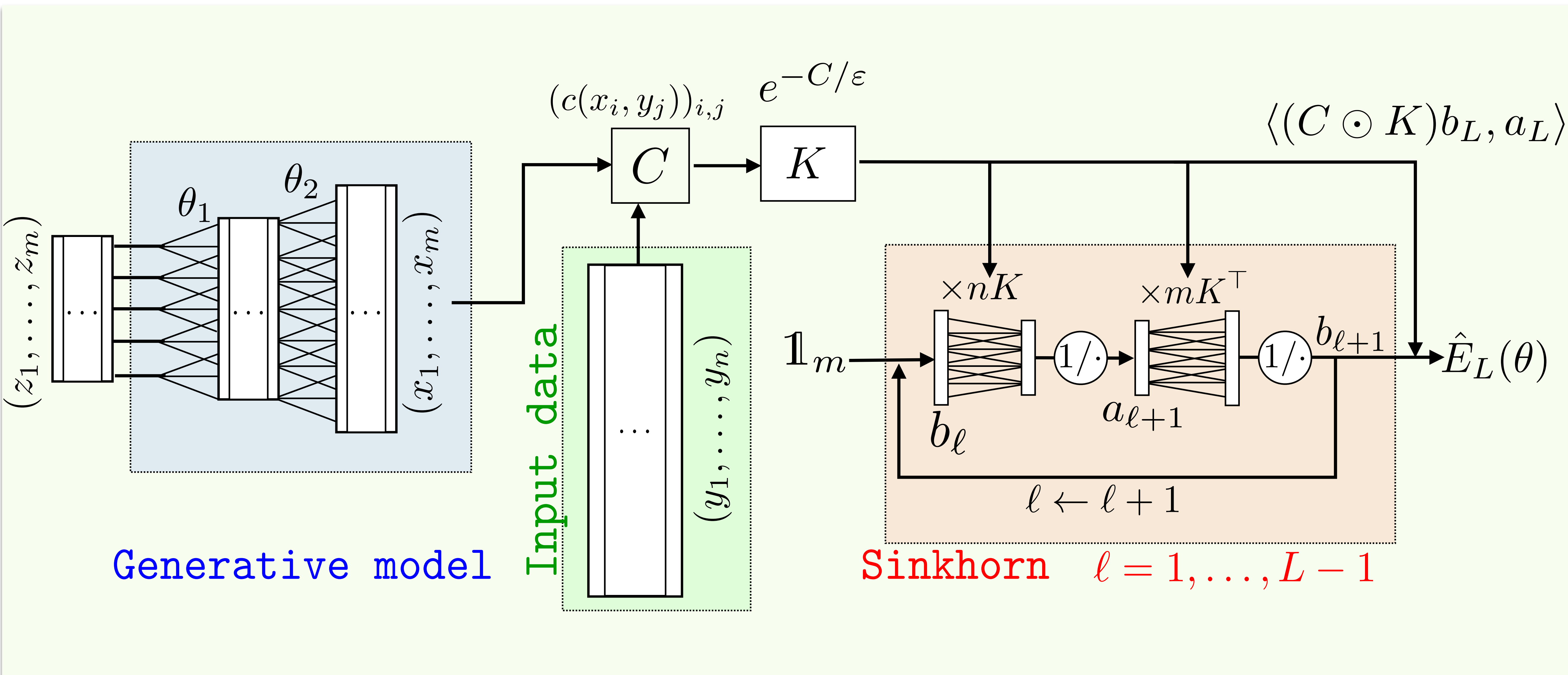
[Arjovsky'17] *(WGAN)* uses a NN to approximate dual solutions and recover gradient w.r.t. parameter

[Bernton'17] *(Wasserstein ABC)*

[Genevay'17, Salimans'17] *(Sinkhorn approach)*

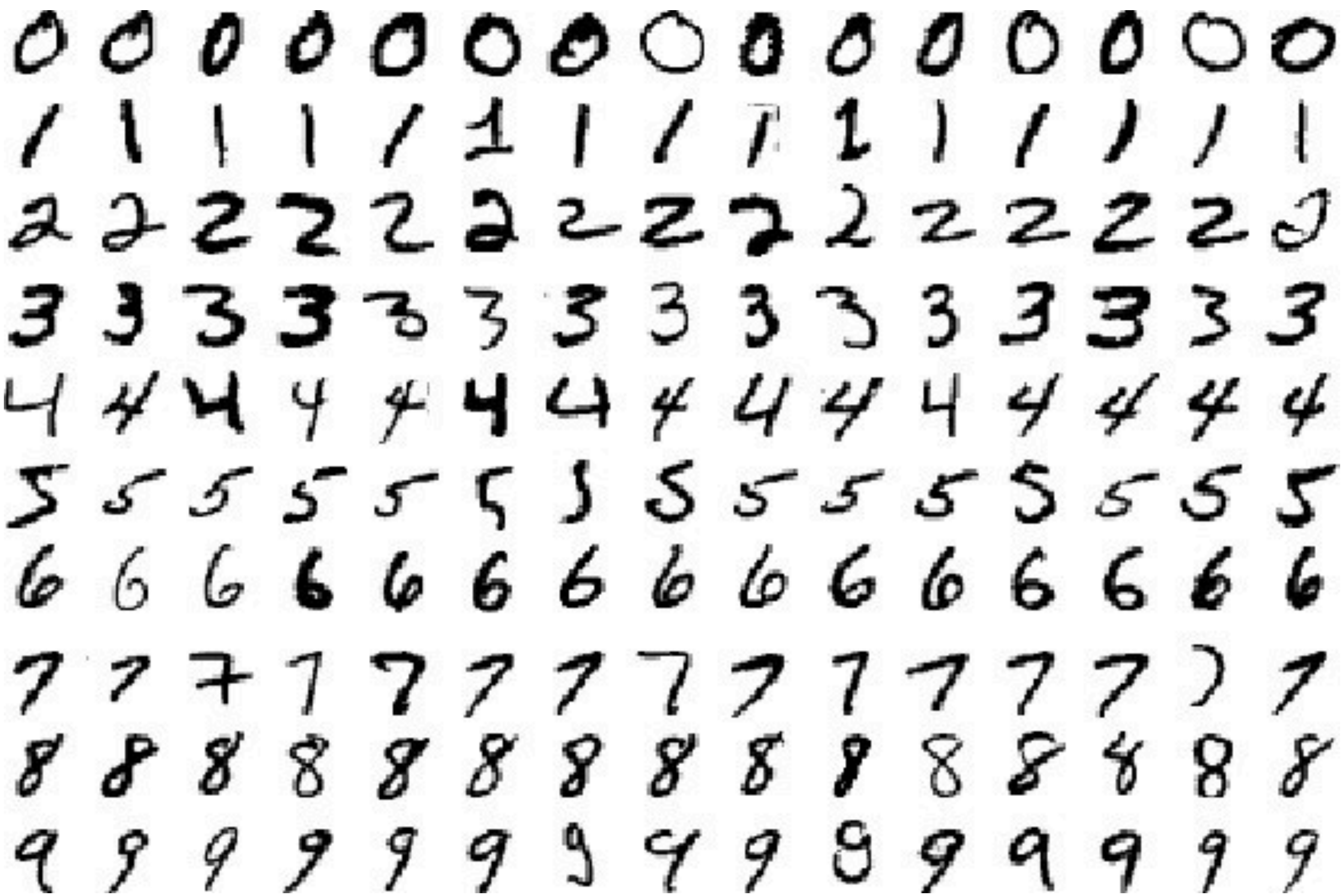
Proposal: Autodiff OT using Sinkhorn

Approximate W loss by the transport cost $\bar{W}_L$ after L Sinkhorn iterations.



[GPC'17]

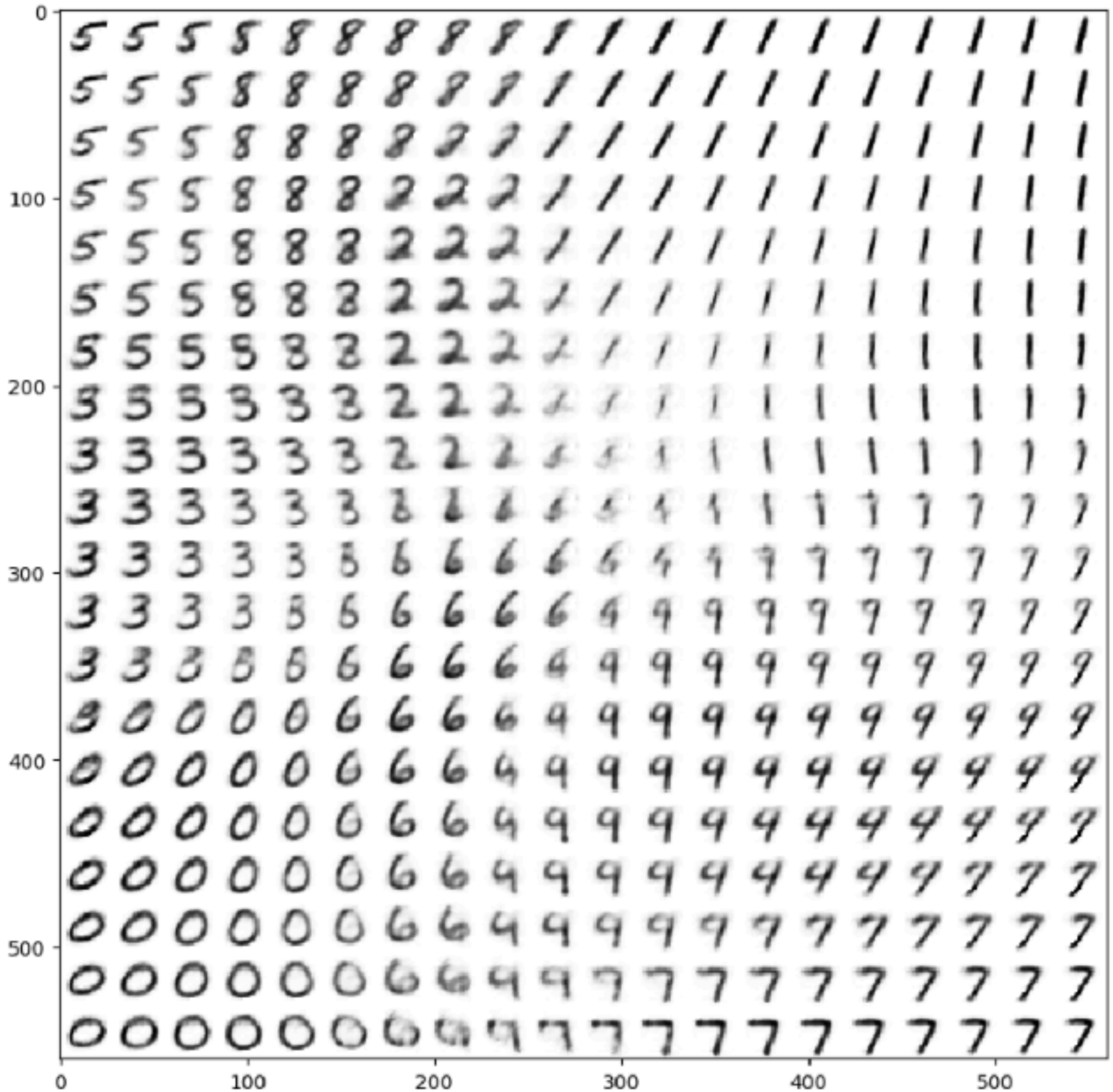
Example: MNIST, Learning $f_{\boldsymbol{\theta}}$



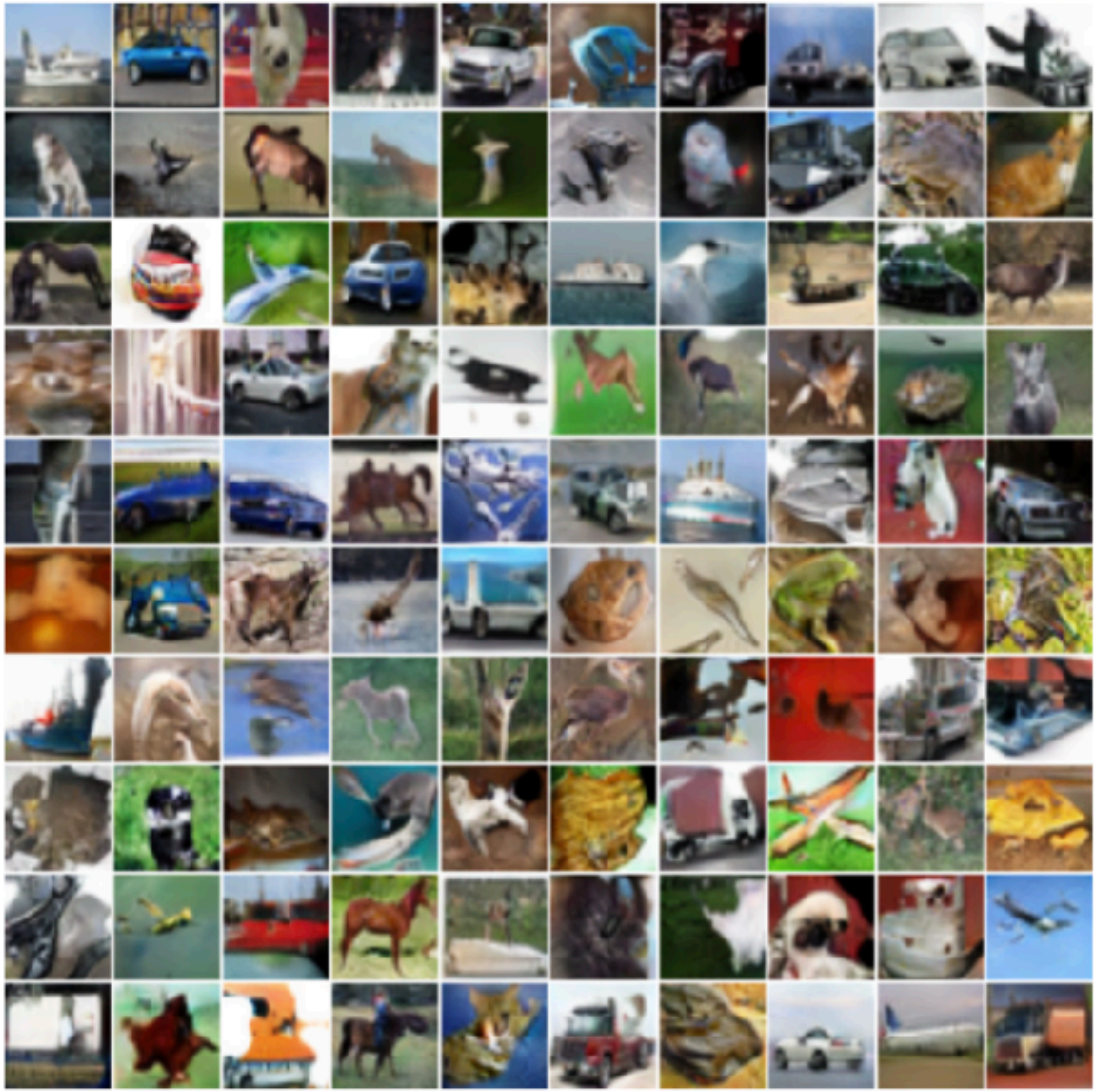
Example: MNIST, Learning f_{θ}

Latent space

$[0, 1]^2$



Example: Generation of Images



arxiv.org/pdf/1710.05488

[Salimans'18]

Example: Generation of Images

arxiv.org/pdf/1710.05488

[Salimans'18]

Concluding Remarks

- *Regularization* is required for OT to work on data.
- If one expects that these tools become widely adopted, they must be *"auto-differentiable"*.
- **Many open problems remain!**

What I could not talk about...

- Very large supply of maths...
- **Statistical** challenges to compute W.
- If **linear assignment** = Wasserstein, then **quadratic assignment** = Gromov-Wasserstein.
- Wasserstein gradient flows (a.k.a. **JKO** flow).
- **Dynamical** aspects of optimal transport
- Transporting vectors and matrices